ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA Y SISTEMAS DE TELECOMUNICACIÓN

PROYECTO FIN DE GRADO

TÍTULO:
DISEÑO DE UN VIDEOJUEGO ORIENTADO A MEJORAR EL PROCESO DE ENSEÑANZA-APRENDIZAJE DE LA LENGUA INGLESA

AUTOR:

CRISTINA VILLAR MIGUELEZ

TITULACIÓN:

GRADO EN INGENIERIA DE SISTEMAS DE TELECOMUNICACIÓN

TUTOR:

VÍCTOR JOSÉ OSMA RUIZ

DEPARTAMENTO:

INGENIERÍA DE CIRCUITOS Y SISTEMAS (ICS)

VºBº

Miembros del Tribunal Calificador:

PRESIDENTA: IRINA ARGÜELLES ÁLVAREZ

VOCAL: VÍCTOR JOSÉ OSMA RUIZ

SECRETARIA: JUANA MARÍA GUTIÉRREZ ARRIOLA

Fecha de lectura: Septiembre 2014

Calificación:

El Secretario,

AGRADECIMIENTOS

Cómo empezar este pequeño apartado de agradecimientos en el que sería tremendamente difícil abarcar la inmensa ayuda que he recibido no solamente durante toda la realización de este proyecto, que ha sido excepcional, sino también a lo largo de mi vida estudiantil en esta escuela. Quería aprovechar esta pequeña oportunidad para recordar a todos aquellos profesores que se han preocupado por el correcto desarrollo de mi vida educativa y sancionar, en pequeña medida, a los que no han estado tan atentos como yo pensaba que estarían, a lo mejor por motivos que yo no llegaba a conocer. Sin embargo, estoy satisfecha de la labor docente pues me ha permitido labrar un futuro en el complicado mundo que las Telecomunicaciones suponen.

En primer lugar quería nombrar a la persona que más me ha ayudado y me ha guiado en este proyecto final, siempre ha velado por el correcto transcurso del proyecto y, lo más importante para mí siempre ha resuelto mis dudas sobre cualquier aspecto, aunque estoy segura de que algunas de ellas no tenían demasiado sentido. Estoy hablando de mi tutor Víctor Osma Ruíz. Al que también quiero darle las gracias por haberme ofrecido un trato tan personal y comprensivo.

Cómo no también mencionar a Nico que siempre ha sido nuestra ayuda en la toma de decisiones en cuanto al apartado de resultados de este proyecto se refiere y en cuanto a proporcionar una tercera opinión en las cuestiones que le planteábamos.

También querría dar las gracias a la sección del departamento de Ingeniería de Circuitos y Sistemas de la Universidad Politécnica de Madrid, y especialmente a Juana María Gutiérrez Arriola por la excelente oportunidad que me proporcionó allá por inicios del mes de septiembre de 2013 y que ha permitido desarrollar este proyecto dentro de un contexto investigador que he podido experimentar y que ha resultado realmente enriquecedor.

Por supuesto voy a mencionar a aquellas personas que me ayudaron a realizar las pruebas del reconocedor, teniendo que repetir muchas veces las mismas palabras y sin apenas quejarse. Esas personas que accedieron a hacer las pruebas por ese sentimiento de amistad tan necesario en nuestras vidas y que nos ayuda a ser mejores personas, no hace falta que los mencione para que sepan que el sentimiento hacia a ellos es mutuo: Mari, Patri, Alberto, Iván y otros amigos que me ayudaron apoyándome y dándome ánimos en los momentos de tristeza: Evita, Cris, Eva, Anna…y también a los compañeros del laboratorio de investigación, aunque seguro que me olvidaré de mencionar a alguno.

Y por último y no menos importante, dar las gracias a mi familia por ser un apoyo constante durante la vida universitaria, y no me refiero al apoyo económico que también, sino a la ayuda emocional que es infinitamente más importante y necesaria. A mi madre, a mi padre y a mi hermano, que también contribuyeron a la realización de las pruebas de este proyecto, que siempre están y no me cabe duda que estarán cuando más lo necesite.

Gracias de verdad a todos los mencionados pues sin ellos la vida hubiese resultado un poquito más dura.

A mi familia y amigos de verdad.

Freud, Sigmund: "Sólo la propia y personal experiencia hace al hombre sabio."

Beethoven, Ludwig van: "Haz lo necesario para lograr tu más ardiente deseo, y acabarás lográndolo."

RESUMEN

Desde que el proceso de la globalización empezó a tener efectos en la sociedad actual, la lengua inglesa se ha impuesto como primera opción de comunicación entre las grandes empresas y sobre todo en el ámbito de los negocios. Por estos motivos se hace necesario el conocimiento de esta lengua que con el paso de los años ha ido creciendo en número de hablantes.

Cada vez son más las personas que quieren dominar la lengua inglesa. El aprendizaje en esta doctrina se va iniciando en edades muy tempranas, facilitando y mejorando así la adquisición de una base de conocimientos con todas las destrezas que tiene la lengua inglesa: lectura, escritura, expresión oral y comprensión oral.

Con este proyecto se quiso mejorar el proceso de enseñanza-aprendizaje de la lengua inglesa en un rango de población menor de 13 años.

Se propuso crear un método de aprendizaje que motivara al usuario y le reportase una ayuda constante durante su progreso en el conocimiento de la lengua inglesa. El mejor método que se pensó para llevar a cabo este objetivo fue la realización de un videojuego que cumpliese todas las características propuestas anteriormente.

Un videojuego de aprendizaje en inglés, que además incluyese algo tan novedoso como el reconocimiento de voz para mejorar la expresión oral del usuario, ayudaría a la población a mejorar el nivel de inglés básico en todas las destrezas así como el establecimiento de una base sólida que serviría para asentar mejor futuros conocimientos más avanzados.

PALABRAS CLAVE

Juegos serios
Juegos educativos
Aprendizaje con juegos
Reconocimiento de voz
Pronunciación
Sistema de tutelaje
Enseñanza de lenguas asistida por ordenador
Gamification

ABSTRACT

Since Globalization began to have an effect on today's society, the English language has emerged as the first choice for communication among companies and especially in the field of business. Therefore, the command of this language, which over the years has grown in number of speakers, has become more and more necessary.

Increasingly people want to master the English language. They start learning at very early age, thus facilitating and improving the acquisition of a new knowledge like English language. The skills of English must be practiced are: reading, writing, listening and speaking. If people learnt all these skills, they could achieve a high level of English.

In this project, the aim is to improve the process of teaching and learning English in a range of population less than 13 years. To do so, an interactive learning video game that motivates the users and brings them constant help during their progress in the learning of the English language is designed.

The video game designed to learn English, also includes some novelties from the point of view of the technology used as is speech recognition. The aim of this integration is to improve speaking skills of users, who will therefore improve the standard of English in all four basic learning skills and establish a solid base that would facilitate the acquisition of future advanced knowledge.

KEY WORDS

Video games
Gamification
Serious games
Educational value of games
Game-based learning
Speech recognition
Pronunciation
Tutoring system
CALL (Computer-Aided Language Learning)
Gamification

TABLA ACRÓNIMOS

TIC:	Tecnologías de la información y la comunicación.
CALL:	Computer-Aided Language Learning (Enseñanza de lenguas asistida por ordenador).
API:	Application Programming Interface (Interfaz de programación de aplicaciones).
CMU:	Carnegie Mellon University (Universidad Carnegie Mellon).
REAP.PT:	REAder-specific Practice Portuguese (Práctica de la lectura del portugués).
UML:	Unified Modeling Language (Lenguaje unificado de modelado).
LGPL:	Library General Public License (Licencia pública general para librerías).
EULA:	End User License Agreement (Acuerdo de licencia de usuario final).
SDL:	Simple DirectMedia Layer (Capa simple de Direct Media).
SFML:	Simple and Fast Multimedia Library (Librería multimedia simple y rápida).
ALLEGRO:	Allegro Low Level Game Routines (Rutinas de juego de bajo nivel Allegro).
HTK:	Hidden Markov Model Toolkit (Herramientas del modelo oculto de Markov).
BASIC:	Beginner's All-purpose Symbolic Instruction Code (Código simbólico de instrucciones de propósito general para principiantes).
BBDD:	Base de datos o banco de datos.

ÍNDICE DE CONTENIDOS

1. INTRODUCCIÓN Y OBJETIVOS 15
2. ESTADO DEL ARTE 17
3. DESCRIPCIÓN DE LAS HERRAMIENTAS DISPONIBLES 25
3.1. INTRODUCCIÓN 25
3.2. ALTERNATIVAS DISPONIBLES 26
3.2.1. Programación del videojuego 26
Framework: 26
SDL 26
PyGame 26
SFML 26
ALLEGRO 27
LibGDX 27
Marmalade 27
XNA 28
Game engine: 28
Andengine 28
Unity 28
Ogre3D 29
3.2.2. Librerías de reconocimiento 30
Speech SDK 5.1 30
Dragon Naturally Speaking 30
CMU SPHINX 31
Julius 32
HTK 32
Voce 32
Librerías AT&T 33
Asset WordDetection 33
4. DESCRIPCIÓN DE LA SOLUCIÓN 35
4.1. Elección del entorno de programación 35
4.2. Elección de la librería de reconocimiento 36
4.3. Descripción de problemas y soluciones propuestas 37
4.3.1. Alternativa librerías AT&T 37
4.3.2. Alternativa asset “WordDetection” 38

5. DISEÑO DEL VIDEOJUEGO 39
5.1. Elementos “gamification” 39
5.2. Diagrama de flujo 40
5.3. Diagramas UML 42
5.4. Matriz de dependencias 48
5.5. Entorno de sistema 51
6. RESULTADOS DEL RECONOCIMIENTO 53
6.1. Grabación de palabras con un único locutor 54
6.2. Grabación de 4 palabras con 4 locutores 56
6.3. Grabación de 4 palabras con diferentes intensidades y posiciones usando 4 locutores 65
6.4. Captura del ruido ambiente 67
7. MANUAL DE USUARIO 69
7.1. Entorno de sistema requerido 69
7.2. Configuración del micrófono 69
7.3. Inicio del videojuego 70
8. CONCLUSIONES Y LÍNEAS FUTURAS 77
8.1. Conclusiones 77
8.2. Líneas futuras 78
9. REFERENCIAS 81
Estado del arte 81
Desarrollo de videojuegos 83
Reconocimiento de voz 84
Líneas futuras 85
Documentación adicional 85

ÍNDICE DE FIGURAS

Figura 1. Diagrama de flujo del videojuego. 41
Figura 2. Diagrama UML general del videojuego. 42
Figura 3. Diagrama UML del gestor de terrenos 2D. 43
Figura 4. Diagrama UML de WordDetection. 44
Figura 5. Diagrama UML de estructura de escenas del videojuego. 45
Figura 6. Diagrama UML de escena 1. 46
Figura 7. Diagrama UML de escena 2. 47
Figura 8. Diagrama UML de escena 3. 47
Figura 9. Ventana de configuración del videojuego. 70
Figura 10. Controles del videojuego. 71
Figura 11. Primera escena del videojuego. 72
Figura 12. Escena de ayuda. 72
Figura 13. Escena 2 con avatar de marciano. 73
Figura 14. Continuación de la escena 2. 74
Figura 15. Escena 2 con cambio de personaje. 74
Figura 16. Escena 3 de reconocimiento de voz. 75

ÍNDICE DE TABLAS

Tabla 1. Comparativa de frameworks y entornos de programación. 36
Tabla 2. Comparativa de librerías de reconocimiento de voz..... 37
Tabla 3. Matriz de dependencias. 50
Tabla 4. Opciones tiempo de captura y frecuencia de muestreo. 54
Tabla 5. Relaciones de *jeans* y *house* con diferentes configuraciones. 55
Tabla 6. Resultados frecuencia de 8192 y tiempo de captura 1s. 55
Tabla 7. Resultados frecuencia 8192 y tiempo de captura 2s. 56
Tabla 8. Puntuaciones locutor interno femenino (femenino 1) al pronunciar jeans 57
Tabla 9. Cálculo de relaciones usando el mínimo. 58
Tabla 10. Mínimos de varios locutores..... 59
Tabla 11. Máximos de varios locutores. 60
Tabla 12. Comparativa mínimo, máximo, media y mediana. 60
Tabla 13. Porcentajes de mejora/empeoramiento con respecto al mínimo..... 61
Tabla 14. Comparativa de bases de datos de 4 y 16 palabras con locutor interno femenino……..…….....61
Tabla 15.Puntuaciones de socks y boots usando base de datos de 4 palabras (A)..... 62
Tabla 16. Puntuaciones de socks y boots usando base de datos de 16 palabras (B).....62
Tabla 17. Puntuaciones al pronunciar socks por locutor femenino1 usando base de datos A y B.....63
Tabla 18. Comparativa de bases de datos de 4 y 16 palabras con locutor externo femenino……..……...…..…63
Tabla 19. Comparativa de bases de datos de 4 y 16 palabras con locutor externo masculino..……...……..…...…..64
Tabla 20. Cálculo relaciones con BBDD 4 y 16 palabras al pronunciar jeans.....65
Tabla 21. Comparativa de bases de datos de 16 y 64 palabras con locutor interno femenino……...……..….....…66
Tabla 22. Comparativa de bases de datos de 16 y 64 palabras con locutor externo femenino……...……..….....…66
Tabla 23. Comparativa de bases de datos de 16 y 64 palabras con locutor externo masculino.....……...……..….....…67
Tabla 24. Puntuación obtenida del ruido.....68

1. INTRODUCCIÓN Y OBJETIVOS

Es necesario resaltar que la lengua inglesa ha conseguido implantarse en la sociedad actual como la primera opción a utilizar en diferentes y variados ámbitos tales como empresarial, económico, mercantil o incluso político. Esto es debido principalmente a un motivo, el fenómeno de la globalización, que hace posible que haya una comunicación continua entre los países del mundo, y permite el intercambio continuo de culturas y mercados. Para que este intercambio se produzca con éxito surge la necesidad de un idioma único y universal que permita la comunicación global, y ese idioma es la lengua inglesa.

En este proyecto se pretende contribuir al diseño de un videojuego que incluya el aprendizaje en las destrezas más importantes de una lengua: la pronunciación, la comprensión oral, la escritura, la gramática y la lectura.

El usuario se enfrentará a diferentes retos en lengua inglesa mediante ejercicios con un nivel adecuado a sus conocimientos dentro de un videojuego entretenido, ameno y divertido que hará al usuario aprender sin que sea una carga para él.

Nuestro proyecto se encuentra, pues, enmarcado dentro de la realización de un proyecto más general y completo que consistirá en el desarrollo de un videojuego en formato aventura gráfica. Dicho videojuego integrará todas las pruebas necesarias para mejorar cada una de las destrezas posibles de aprendizaje de una lengua, en nuestro caso concreto es la lengua inglesa. Las destrezas que se mejorarán con este tipo de aprendizaje serán: lectura, escritura, comprensión oral y expresión lectora. Por lo tanto la realización de este proyecto permite avanzar hacia el objetivo último y final de un trabajo de mayor envergadura.

Principalmente el proyecto que se presenta describirá una forma de integrar el reconocimiento de voz en un videojuego didáctico de pequeño tamaño que permitirá al usuario mejorar la pronunciación en la lengua inglesa. Se pretende profundizar en el estudio de las alternativas disponibles de librerías de reconocimiento de voz inglesa y en su aplicación en plataformas de desarrollo de videojuegos. Se buscarán soluciones de aplicación de estas librerías de reconocimiento en ejercicios sencillos que pueda realizar el usuario al que se pretende ofrecer el servicio de aprendizaje.

El objetivo principal de este proyecto es, pues, diseñar un videojuego que permita al usuario mejorar sus habilidades de expresión en lengua inglesa.

Para ello debemos cubrir varios objetivos menores:

1. Estudiar los diferentes lenguajes de programación de videojuegos existentes y seleccionar el más adecuado para nuestro propósito.

2. Estudiar los tipos de ejercicios que sirven para mejorar la gramática, la comprensión lectora, la comprensión oral o la escritura.

3. Determinar el nivel de los ejercicios diseñados y organizarlos.

4. Diseñar los gráficos del videojuego.

5. Estudiar diferentes librerías sobre reconocimiento de habla inglesa, haciendo para ello numerosos ejemplos y pruebas sencillas.

6. Realizar pruebas que evalúen la calidad del reconocimiento de la voz.

7. Diseñar y desarrollar el videojuego, según los ejercicios mencionados anteriormente y haciendo uso de las librerías de reconocimiento más óptimas.

2. ESTADO DEL ARTE

El contexto actual que enmarca la época en la que nos encontramos, está plenamente inmerso en el uso de las nuevas tecnologías de la información y la comunicación (TIC) [1]. La proliferación de la utilización de nuevos dispositivos digitales tales como teléfonos móviles u ordenadores, y el desarrollo de la red de redes, Internet, hacen posible que la aplicación de las TICs alcance multitud de campos variados tales como el comercial, empresarial o militar. Sin embargo si nos centramos en las necesidades individuales de las personas entran en juego otros campos diferentes y no por ellos menos importantes como la educación, el ocio y la sanidad. Con el paso del tiempo se hace más evidente la presencia y la importancia de las TICs en la actualidad, observándose un ritmo de crecimiento exponencial debido a la gran cantidad de soluciones y alternativas que ofrecen. Se han convertido en una herramienta imprescindible en el transcurso de nuestra vida, tanto que se ha acuñado un término para el conjunto de la población que ha conocido o ha crecido en el entorno de las TICs: "Net Generation" [2]. Las nuevas generaciones de niños y adolescentes se están desarrollando íntegramente en un contexto completamente computerizado, repleto de nuevas tecnologías digitales.

Por otro lado hay que señalar que precisamente debido al rápido crecimiento de estas tecnologías, se hace aún más necesaria la realización de una evaluación sobre el grado de utilidad que presentan, diferenciando entre los distintos campos: educativo, empresarial ó cualquiera que pudiera hacer uso de dichas tecnologías.

Uno de los campos más importantes en los que es necesario profundizar para conocer cómo se desarrollan las TICs, y cuáles son sus resultados, es la educación. En este sector es imprescindible la evaluación de la calidad de las tecnologías digitales aplicadas ya que está aumentando el número de recursos disponibles que fusionan educación y TICs, y no siempre se hace de la mejor manera posible debido a la rapidez de su creación.

Algunos de los recursos educativos disponibles basados en las TICs que más se están realizando son aplicaciones para teléfonos móviles, webs de aprendizaje de todo tipo de destrezas, plataformas de enseñanza online y videojuegos educativos de diversa índole: online, para móviles ó para PCs.

Sin embargo hay un tipo de recurso determinado que está provocando mucho interés, el videojuego educativo. Diferentes investigaciones lo han analizado y estudiado [3] [4] [5] [6] [7]. De las investigaciones anteriores se extrae que el videojuego educativo está enmarcado dentro de los juegos serios y que este tipo de juegos tienen una gran utilidad en el aprendizaje de nuevas destrezas. Concretamente los autores de [5] hacen una comparativa de las características de los videojuegos y los juegos serios. La definición de juegos serios proporcionada en [5] exponía que se trataban de un conjunto de videojuegos diseñados específicamente para alcanzar un objetivo útil, como puede ser el hecho de aprender un nuevo idioma.

Además, declaran que la mayoría de los videojuegos no son efectivos en el proceso de aprendizaje y que están destinados al mero entretenimiento, por otro lado destacan que los juegos serios sí que son útiles en este campo. A pesar de que estas declaraciones están aún pendientes de comprobación de una forma experimental, pues los casos prácticos evaluados fueron escasos, se puede extraer que los juegos serios motivan al usuario y que esta motivación ayuda a cumplir el objetivo para el cual el juego fue diseñado, en nuestro caso el aprendizaje.

No solo los juegos serios motivan al usuario, sino también las nuevas tecnologías aplicadas a la educación. Incrementan el nivel de atención y fomentan la iniciativa a asimilar nuevas destrezas en un contexto diferente, que no es el típico de un aula con libros sino que es divertido y entretenido [8].

El origen del uso de juegos que ayudan al aprendizaje ya se analizó mucho tiempo atrás [9], pero es actualmente cuando están proliferando los juegos educativos y especialmente los videojuegos educativos y por eso como ya se dijo anteriormente, es necesario fijarse en cómo y cuánto ayudan dichas herramientas al aprendizaje.

Para saber si un videojuego cumple con los requisitos mínimos de juego serio, conviene explicar primero diferentes términos relacionados con los juegos cuyo objetivo principal no es el mero entretenimiento. El empleo de las dinámicas y mecánicas de juegos para adquirir hábitos y alcanzar determinados objetivos es lo que llamamos “gamification” [7]. Algunas de los elementos de juegos que más se emplean en “gamification” se definen a continuación:

- Puntos: se acumulan durante las distintas actividades del juego.
- Insignias: representaciones visuales de recompensas que se pueden recolectar durante el juego.
- Tabla de líderes: lista de los mejores jugadores.
- Barra de progreso: determina el estado actual para completar un objetivo final.
- Gráficos de actuación: muestra información de las actividades de otros jugadores.
- Misiones: pequeñas tareas que se deben completar para terminar el juego.
- Historias con un significado completo: videojuego desarrollado en un contexto que tiene principio y fin y que va evolucionando conforme el usuario va avanzando en el videojuego.

- Avatares: diferentes representaciones del jugador principal.

- Desarrollo de perfiles: referido a que cada avatar posee un conjunto de actitudes y características.

Cada uno de los elementos anteriores aporta diferentes mecanismos de motivación al juego, necesarios para el buen aprendizaje. A continuación se exponen las relaciones entre los elementos de juego y las formas de motivación [7]:

- Puntos: refuerzos positivos inmediatos.

- Insignias (para conseguirlas son necesarios diferentes niveles de complejidad) :
    - Representan la necesidad por el éxito del jugador y provocan una gran sensación de motivación.
    - Son símbolos de estado virtual que capturan la atención del jugador.
    - Permiten compartir experiencias y actividades del juego y por tanto son los responsables de provocar un sentimiento de afiliación con el juego.
    - Poseen la función de establecer un objetivo.
    - Potencian el sentimiento de competencia dentro del juego, que ayudará a que los usuarios se mantengan motivados.

- Tabla de líderes:
    - Motivan al jugador para poder estar en la primera posición.
    - Mantiene una competencia entre jugadores.
    - Como también recogen una puntuación de equipo, ayuda a la colaboración entre usuarios.

- Barra de progreso y gráficos de actuación:
    - Permiten obtener feedback.
    - Destacan los objetivos principales a realizar.
    - Potencian el seguimiento del juego en dirección a un objetivo.

- Historias con un completo significado:
    - Incrementan el interés de los jugadores en el contexto en el que se desarrolla el juego.
    - Promueven el sentimiento de autonomía permitiendo al jugador elegir entre diversas opciones de historias.
    - Incrementan los sentimientos positivos hacia el juego.

- Avatares y desarrollo de perfiles:
    - Eligiendo distintos avatares se tienen distintas formas de jugar, lo que aumenta la autonomía.
    - Sentimientos positivos y emocionales van asociados al desarrollo del avatar durante el juego.

Los videojuegos que aplican los elementos de juego anteriormente definidos en el contexto de "gamification" y cuyo objetivo no es solo de ocio, son también llamados "juegos serios" [10].

Algunos autores creen que este tipo de videojuegos son eficaces en el proceso de aprendizaje y así lo explican en [3] [4]. Mientras que otros explican los pasos a seguir cuando se quiere realizar una evaluación de los videojuegos educativos [5]. Este tipo de evaluación consiste en comparar a dos grupos de alumnos distintos: uno es el grupo que usa el videojuego durante su instrucción y el otro el que no lo hace, sustituyendo este por pruebas y ejercicios típicos. A estos grupos se les haría una prueba previa al uso del videojuego y una prueba posterior y se compararían los resultados, obteniendo así una medida de la eficacia de los videojuegos en el ámbito educativo. Esta evaluación no acabaría aquí ya que acorde a la opinión de los autores de [5] sería necesaria una prueba a largo plazo que midiese la retención de conocimientos en cada uno de los dos grupos.

El éxito o fracaso de un videojuego en el ámbito educativo depende en cierta medida de la forma en la cual se ha desarrollado y por eso hay que tener en cuenta una serie de consideraciones y criterios iniciales. Un buen videojuego educativo debe permitir una comunicación constante entre el usuario y el instructor, es decir, ofrecer un feedback continuo entre ambos, enseñar nuevos conocimientos, permitir al usuario explorar libremente el videojuego, provocar curiosidad en el usuario, poseer un nivel de dificultad medio para motivar al usuario, seguir un método pedagógico determinado, ofrecer recompensas cuando los ejercicios han sido realizados correctamente, establecer una serie de objetivos claros y principales y sobre todo promover el aprendizaje semipresencial y online[6]. Además, si los videojuegos se basaran en teorías educacionales, los resultados directos sobre el usuario serían más útiles y exactos. Algunas de estas teorías son:

- Constructivismo: el usuario forma su conocimiento interactuando con el medio, en nuestro caso actuando con los elementos del videojuego [11].

- Conductismo: si el usuario del videojuego cumple bien las tareas requeridas recibirá un premio y por tanto se reforzará este tipo de comportamiento [12].

- Teoría de Scaffolding: es una técnica consistente en crear una actividad educativa a partir de subtareas. Aplicando esta técnica al campo del videojuego, se tendrían varias fases intermedias en el juego que el usuario tendría que pasar para completar un hito educativo completo [13].

Según el género en el que se encuentre definido el videojuego educativo, ofrecerá una serie de características de aprendizaje u otras. Los juegos de estrategia y las aventuras gráficas donde el usuario toma el control de distintos personajes son los más idóneos en el proceso de aprendizaje [6]. Concretando aún más, los juegos de estrategia permiten la comunicación entre jugadores, un concepto de inmersión total en el juego, un feedback al ir progresando, permiten jugar de una forma no lineal ya que se puede decidir lo que hacer en cada momento y aceptan diferentes soluciones a un mismo problema. Por otro lado los aventuras gráficas enseñan nuevos conocimientos a través del juego, provocan curiosidad en el usuario, ofrecen un tipo de juego basado en recompensas y ofrecen feedback durante el progreso completo del juego[6][14].

Sabiendo que el objetivo principal de los videojuegos educativos es ser eficaces y conseguir que los usuarios aprendan nuevas destrezas, es condición indispensable que sean atractivos y posean una buena jugabilidad pero sin caer en el error de perfeccionar solamente la parte gráfica y visual [15] ya que los recursos sonoros y animaciones dentro del videojuego no es lo que más ayuda al proceso de aprendizaje. Se encuentran también estudios [15] que certifican que la aplicación más destacada de este tipo de videojuegos es despertar el interés del usuario sobre un nuevo tema, y prepararlo para poder adquirir un conocimiento mayor a través de otros medios como por ejemplo una clase con un profesor o una tutoría más personalizada. También podrían usarse como complemento educativo para asentar los conocimientos adquiridos en una clase previa, es decir, utilizarlos como una herramienta curricular.

Según todo lo expuesto anteriormente, parece claro que una de las mejores aplicaciones que pueden tener los videojuegos educativos es mejorar el aprendizaje de una nueva lengua extranjera, lo que apoya el objetivo de este proyecto.

Este tipo de videojuegos educativos que enseñan una lengua entran en la categoría de "Enseñanza de Lenguas Asistida por Ordenador", conocida como CALL por sus siglas en inglés (Computer-Aided Language Learning). La ayuda que proporcionan tanto los ordenadores como los recursos electrónicos en el proceso de enseñanza mejora la motivación de los alumnos y permite el aprendizaje semipresencial y autónomo [16]. Aunque para que estas técnicas de enseñanza digital tengan éxito es necesario que los instructores encargados las dominen perfectamente para poder así transmitir los conocimientos a sus alumnos sin ningún percance [16]. En [16] se expone que actualmente son necesarios más recursos software de calidad para aplicar en su totalidad la técnica CALL.

Acorde a un estudio del año 2011 [17] los profesores que impartían una lengua extranjera pensaban que el uso de los videojuegos educativos tenía un mayor impacto en su campo que en otras materias. Además se ha podido descubrir que existe una gran variedad de videojuegos que ayudan al aprendizaje de una nueva lengua, tales como [18] [19] [20]. El videojuego TLCTS (Tactical Language and Culture Training System) [19] permite a los militares de USA aprender los idiomas de las zonas a las que viajaban, el primer prototipo estaba centrado en la destreza oral del Árabe Levantino y el segundo en aprender la cultura y el lenguaje Árabe-Iraquí. Este tipo de juego serio fue implementado en el año 2010. Otro ejemplo de videojuego para aprender una lengua nueva es el llamado REAP.PT (REAder-specific Practice Portuguese) que enseña portugués a través de ejercicios de vocabulario y gramática [18]. Finalmente se estudió un último ejemplo de juego serio cuyo objetivo era común al anterior, aprender un nuevo idioma. Este proyecto se relaciona con la realidad aumentada, y concretamente ayuda al aprendizaje del español teniendo como marco geográfico de la realidad virtual a la ciudad de Madrid. Los usuarios pueden interactuar con sus teléfonos móviles y con el medio que les rodea para evolucionar en sus conocimientos [20].

Sin embargo, faltan muchos puntos por definir para que estos videojuegos sean realmente eficaces en el proceso aprendizaje. Pues son necesarios estudios de investigación en profundidad para determinar si el aprendizaje es el correcto o no, aunque existen algunos que dan respuesta a esta cuestión usando simplemente las opiniones de los jugadores [18] [19] [20] [21] [22].

El aprendizaje de la lengua inglesa se ha convertido en una materia obligatoria en el sistema educativo español y no siempre se han obtenido buenos resultados en niños de corta edad o adolescentes. Es necesario formar una base sólida de conocimiento de este idioma a edades tempranas para poder construir unas destrezas más sólidas en el futuro. La lengua inglesa es el idioma técnico por excelencia y por tanto un conocimiento básico y necesario en la sociedad globalizada en la que vivimos. Por esto se decide que el proyecto tenga como objetivo el aprendizaje de la lengua inglesa en todas las destrezas posibles.

Teniendo en cuenta que el objetivo del proyecto es crear un videojuego que ayude al aprendizaje de la lengua inglesa, se tiene que hacer un análisis de la situación actual de los videojuegos relacionados directamente con este tema. La mayoría de los videojuegos encontrados están formados por un conjunto de mini juegos destinados cada uno a mejorar alguna parte de la lengua [21] [23] [24]. Mingoville® [23], por ejemplo, es una plataforma online de enseñanza de la lengua inglesa en un contexto global. Pretende unificar el aprendizaje obtenido en el centro educativo junto al obtenido en situaciones extraescolares. Es una plataforma que se adapta continuamente a las necesidades de padres, alumnos y profesores. Con el uso de herramientas como Mingoville® se proporciona una atención especializada y un feedback inmediato del progreso de aprendizaje para cada usuario. Se consigue así que tanto padres y profesores estén informados en todo momento del aprendizaje de los alumnos. Como ya se dijo anteriormente Mingoville® ofrece un aprendizaje basado en mini juegos. Esta estructura de videojuego en pequeñas y cortas pruebas puede ser atractiva en edades tempranas, sin embargo pierde atractivo según se avanza en la edad de los niños, cerca de la adolescencia o adolescentes [23]. Por eso un género de videojuego que atraerá más al público adolescente es la aventura gráfica, donde prima la libertad del usuario para realizar una serie de pruebas repartidas por el escenario.

Se han encontrado algunos videojuegos a modo de aventura gráfica que se asemejan a la forma en la que se quiere desarrollar este proyecto [25] [26]. El primero de ellos llamado Pulitzer® es un videojuego para plataforma PC online, que incluye gramática, vocabulario, comprensión oral y lectura de lengua inglesa [25]. El segundo se llama PlayEnglish® y es un videojuego desarrollado para PlayStation® que incluye ejercicios de inglés mezclados con el contexto de la historia para hacerlos más entretenidos, además su sistema de aprendizaje viene avalado y revisado por Vaughan Systems® [26]. En ninguno de ellos se incluye un reconocedor de voz que proporcione un mecanismo para la mejora de la expresión oral del usuario.

También se ha hecho una investigación previa sobre los posibles sistemas que incluían reconocimiento de voz para saber qué metodologías se aplicaban para llevarlo a cabo y tener un conocimiento general del procedimiento. Una de las opciones que se encontró fue NativeAccent™ [27], un instructor de la pronunciación de lengua inglesa para personas no nativas. Sus orígenes están en los estudios de Carnegie Mellon pero el proyecto real se realizó en la compañía Carnegie Speech. Este tutor virtual guarda el progreso de la mejora en la pronunciación, creando un sistema de aprendizaje personalizado y adecuado a las necesidades de cada usuario. Cuando se detecta un error en la pronunciación del usuario, se incluye la información necesaria para subsanarlo, dibujos de la posición de los labios y la lengua junto a numerosas explicaciones, es decir un feedback hacia el usuario [27]. Otro ejemplo de sistema que incluía el reconocimiento de voz es el videojuego de

Interactive Drama Inc (IDI), el cual pone a disposición su software para mejorar el aprendizaje de la lengua Árabe de un conjunto de militares de la armada de U.S.A. El videojuego consiste en crear personajes virtuales los cuales ofrezcan experiencias reales relacionadas con conflictos bélicos. Los usuarios deben entender lo que se expone en cada situación y a la vez expresarse usando el idioma árabe para cumplir los objetivos del videojuego. Se debe contemplar como una forma de simular una conversación en posibles situaciones reales [28]. La última alternativa encontrada que incluye el reconocimiento en un videojuego es el proyecto desarrollado por Army Research Institute (ARI). Consiste en crear un Microworld, tipo de tecnología educativa basada en el aprendizaje constructivista, en el que el usuario interactúa con los elementos del escenario virtual mediante la voz. El objetivo de este videojuego es aprender a expresarse en la lengua árabe moderna. En un principio la tecnología propuesta se prueba en el ámbito militar para después ponerla a disposición de cualquier persona que necesite aprender árabe. Para el reconocimiento de voz se usó el software de Dragon Systems, el cual permite reconocer palabras de forma individual, y si se presenta una frase larga, ésta debe fragmentarse primero para ser reconocida [29].

También se ha tenido en cuenta a la hora de evaluar el marco actual un hecho muy significativo, la gran cantidad de fondos europeos que son destinados al apoyo del desarrollo de videojuegos enmarcados en el ámbito descrito anteriormente de “gamification”. Y el gran número de investigaciones sobre este tema que se están desarrollando actualmente.

Respecto al futuro de las aplicaciones que podrían surgir aplicando el concepto de “gamification” se han realizado diferentes propuestas que tendrán que seguir siendo investigadas para realizarlas de la mejor manera posible [30]. Algunas de estas propuestas son: la realidad aumentada, el control de gestos del usuario o la detección de emociones. La realidad aumentada podría apoyar de manera muy significativa al proceso educativo, viviendo en primera persona y de forma virtual desde acontecimientos históricos pasados hasta una conversación en cualquier idioma con un usuario virtual. Sería una experiencia muy enriquecedora y motivadora para el usuario que la utilizara. Por otro lado, el control de gestos del usuario o la detección de emociones podrían servir de apoyo a la implicación completa del usuario en un videojuego educativo.

Teniendo en cuenta los puntos desarrollados anteriormente el objetivo general de este proyecto es desarrollar y posibilitar la incorporación del reconocimiento de voz a un videojuego cuyo propósito final es mejorar el aprendizaje en la lengua inglesa de niños de 6º primaria. El reconocimiento de voz se encargará de evaluar la pronunciación de los niños para mejorar su habilidad en esta destreza a la vez que ofrecerá un feedback inmediato a los usuarios y a los evaluadores de los usuarios para poner de manifiesto las posibles carencias o logros.

El objeto diferenciador con respecto a las alternativas existentes será que este videojuego se desarrollará como una aventura gráfica con diversas pruebas para el usuario, dónde se integrará la enseñanza de una nueva lengua junto con elementos motivadores que harán de este videojuego la herramienta idónea para el apoyo del aprendizaje de la lengua inglesa. Todo ello llevado a cabo bajo el amparo del programa de estudios regulado por el Ministerio de Educación para el 2ºciclo de educación primaria.

3. DESCRIPCIÓN DE LAS HERRAMIENTAS DISPONIBLES

3.1. INTRODUCCIÓN

Una vez se tienen definidos los objetivos del proyecto se procede a determinar las posibles alternativas existentes actualmente para llevarlo a cabo.

Para empezar se necesita una herramienta específica que sirva como base a la programación de todos los elementos relacionados con el diseño gráfico y renderización del videojuego. Por eso se realizó una búsqueda intensiva de los diferentes ecosistemas que permitían conseguir dicho objetivo y se obtuvo una gran variedad de opciones que fueron estudiadas para poder realizar la elección más adecuada a las necesidades del proyecto, un videojuego de calidad, atractivo al usuario y fácilmente exportable a cualquier plataforma de uso extendido. Entre las opciones para poder desarrollar un videojuego podemos encontrar dos tipos: “game engine” y “game framework”. Las diferencias de unos y otros son las siguientes:

- Game engine: es una herramienta de alto nivel que se basa en el uso de un conjunto de librerías de programación de videojuegos. No se centra en los detalles inferiores de creación de videojuegos.

- Game framework: es un conjunto de librerías con las cuales el desarrollador llevará a cabo la programación del videojuego. Permite tener un control más exhaustivo de todos los procesos que forman parte del videojuego.

Una vez elegida la herramienta de desarrollo del videojuego se empezará a investigar sobre las formas posibles de integración del reconocimiento de voz en el videojuego. En esta fase habrá que tener en cuenta la elección realizada anteriormente, pues influirá notablemente en la forma de desarrollo del reconocedor. Tiene que existir una completa compatibilidad entre videojuego y reconocedor, sin ningún error de funcionamiento admisible.

3.2. ALTERNATIVAS DISPONIBLES

3.2.1. Programación del videojuego

Framework:

- **SDL(SIMPLE DIRECTMEDIA LAYER)**

Se trata de un conjunto de bibliotecas desarrolladas en lenguaje C que permiten realizar dibujos en dos dimensiones y controlar los efectos de sonido integrables [31].

Originalmente fueron creadas por Sam Lantinga para el desarrollo de videojuegos en la plataforma GNU/Linux.

SDL cuenta con un conjunto de wrappers (subrutinas que permiten la compatibilidad entre lenguajes) destinados a poder usarse con otros lenguajes de programación como C++, Ada, C#, BASIC, Erlang, Lua, Java y Python(el wrapper para este lenguaje se llama Pygame).

Son compatibles con una gran variedad de plataformas: Microsoft Windows, GNU, Linux, Mac OS y QNX.

También es interesante destacar que las herramientas proporcionadas por SDL se distribuyen bajo licencia de software libre, llamada Lesser General Public License (LGPL).

- **PYGAME**

Biblioteca en lenguaje Python que permite la creación de videojuegos en dos dimensiones (2D).

Funciona como interfaz de las bibliotecas SDL (como se dijo anteriormente Pygame es el wrapper que permite usar SDL en el lenguaje Python).

Se caracteriza por ser de fácil implementación. Además, es compatible con múltiples plataformas [32].Sin embargo posee algunas desventajas:

- El soporte de gráficos por hardware es limitado (usa muchos recursos del procesador).
- Python es un lenguaje interpretado (en los juegos que requieran grandes cálculos la velocidad se verá reducida).
- Único soporte para 2D.
- Necesidad de bibliotecas adicionales para el control del sonido.

- **SFML(SIMPLE AND FAST MULTIMEDIA LIBRARY)**

Se trata de un conjunto de librerías gráficas de programación de videojuegos, escritas en C++, pero también disponibles para C, Python, Ruby, OCaml y D [33] [34].

SFML ofrece una alternativa a la biblioteca SDL, usando un enfoque orientado a objetos.

Es una biblioteca multimedia que permite al usuario crear videojuegos y programas interactivos.

Las características principales de este lenguaje son:

- API práctica y reducida.
- Gran control de dispositivos de entrada.
- Buen soporte de audio.

- **ALLEGRO(ALLEGRO LOW-LEVEL GAME ROUTINES)**

Se trata de un conjunto de librerías realizadas en C/C++ que facilitan el manejo multimedia para desarrollar juegos y aplicaciones.

Hay alternativas disponibles para usar ALLEGRO junto a lenguajes de programación como Python, D, Pascal y Lua.

Tiene una implementación basada en el control de OpenGL o Direct3D.

Es multiplataforma, compatible con Windows, Mac, GNU/Linux y Unix y también para plataformas usadas por los teléfonos móviles como iPhoneOS y Android.

En cuanto a la licencia, se permite distribuir libremente y sin coste.

Son librerías especializadas en permitir gráficos 2D [35].

- **LIBGDX**

Es un framework de desarrollo de juegos escrito en Java.

Las plataformas soportadas son: Windows, Linux, MacOSX, Android, iOS, Javascript/WebGL (GWT),Blackberry y HTML5 [36].

Las ventajas del uso de este framework son las siguientes:

- Soporte 2D o 3D .
- Mucha documentación, tutoriales y ejemplos de código.
- Actualizaciones de forma periódica.
- Maneja Audio, input (usuario), física, matemática, archivos varios.
- Utilidades de álgebra lineal y geometría
- Te da un acceso más fácil a las librerías de bajo nivel.
- Se encuentra bajo licencia de Apache 2.0.

Sin embargo el soporte para realizar juegos en 3D está aún en desarrollo.

- **MARMALADE**

Es un conjunto de librerías de desarrollo de aplicaciones interactivas que poseen las siguientes características [37]:

- Lenguaje de programación de desarrollo: C++.
- Compatibilidad multiplataforma: iOS y Windows.
- Posibilidad de gráficos y animaciones 3D.
- Facilita el desarrollo de videojuegos.

- Incluye módulos para la simulación de la física en videojuegos.

- **XNA**

Se trata de una API para el desarrollo de videojuegos que está escrita en C#, y utiliza la licencia EULA (End User License Agreement).

Esta API permite su uso con plataformas relacionadas con Microsoft como Microsoft Windows, Xbox 360, Zune y Windows Phone 7[38].

Ventajas

- C# es un lenguaje moderno y eficaz. Hereda toda la potencia de C++, de una forma sencillo.
- Tiene detrás toda la plataforma .NET y sus características.
- C# es un lenguaje compilado, por lo tanto aporta más velocidad.
- XNA tiene soporte para 2D y 3D.
- El framework XNA está construido sobre directX.
- Facilidad de aprendizaje, se trata de montar diferentes módulos para poder programar.
- Posibilidad de publicación de juegos propios para Xbox 360 y Microsoft Phone.
- Licencia EULA.

Desventajas

- No es libre.
- No soporta Linux ni MAC.

Game engine:

A continuación se presentarán las alternativas más importantes y destacadas de motores de desarrollo destinados a la creación de videojuegos.

- **ANDENGINE**

Las características de Andengine son las siguientes [39]:

- Es un motor libre de desarrollo de videojuegos.
- Está destinado al uso con la plataforma Android.
- Lenguaje de programación: java.
- Tiene soporte, únicamente para 2D.
- Utiliza las librerías de nivel inferior de gráficos OpenGL.
- Posee un módulo para la física basado en box2D que permite crear movimiento en el videojuego.

- **UNITY**

Es un motor de desarrollo de videojuegos multiplataforma.

Unity soporta 3 lenguajes: javaScript, C# y Boo.Sin embargo se pueden encontrar plugins para usar otros lenguajes como C/C++, Objective-C o LUA.

Tiene licencia propietaria, con una versión gratuita y otra de pago por unos 400$.

Permite crear juegos para un elevado número de plataformas como Windows, OS X, Linux, Xbox 360, PlayStation 3, Wii, Wii U, iPad, iPhone, Android y Windows Phone.

El motor gráfico que utiliza es Direct3D (Windows), OpenGL (Mac, Linux), OpenGL ES (Android, iOS), y otras APIs propietarias (Wii).

Es fácil pasar de la programación en plataforma 2D a plataforma 3D.

Posee una excelente calidad en los gráficos.

Posee completa compatibilidad con la herramienta de creación de modelos de gráficos en 3D de licencia libre, Blender[40].

Además cabe destacar la ingente información que posee Unity en la web [41] [42] [43].

- **OGRE3D**

Es un motor gráfico 3D escrito en lenguaje de programación C++.

Sus bibliotecas facilitan la programación pues evitan tener que utilizar directamente librerías gráficas básicas como OpenGL y Direct3D.

El motor es de software libre, licenciado bajo MIT (Massachusetts Institute of Technology) y con una comunidad muy activa [44].

Ventajas

- Permite el uso de técnicas gráficas muy variadas.
- Sistema de scripting de materiales atractivo, ya que incluye numerosos recursos para visualizarlos de la mejor forma posible.
- El sistema jerárquico de representación de los datos en la escena es de tipo grafo, siendo éste muy claro y orientativo.
- Las funciones usadas son muy parecidas a las de OpenGL o DirectX, aunque facilitando la lógica de control de éstas. Además se puede escoger la API con la que se quiere renderizar, OpenGL o DirectX .
- La aplicación es exportable a cualquier plataforma con pocos cambios e incluyendo la adaptación a los tres sistemas operativos globales: Windows, Linux y Mac/OS.

Inconvenientes

- El principal problema es la incompatibilidad para exportar imágenes modeladas en 3D desde el programa Blender al motor gráfico Ogre, ya que las formas de exportación presentan numerosos inconvenientes.
- Desorden en la información disponible de Ogre3d en la red (es difícil encontrar la información que realmente se está buscando).

3.2.2. Librerías de reconocimiento

- **Speech SDK 5.1**

Se trata de un conjunto de interfaces de programación de código nativo (API) dedicadas a tareas de reconocimiento de voz y a la transcripción de voz a texto. Está escrito en C# y en C++ [45].

Con esta librería se puede adquirir y controlar la entrada de voz, capturar la información de los eventos generados por el reconocimiento de voz, y configurar y gestionar los motores de reconocimiento de voz.

Soporta un total de 26 idiomas distintos entre los que están el español y el catalán.

Funcionamiento interno de Speech SDK:

Al principio se procesa el flujo de audio de entrada, aislando segmentos de sonido y convirtiéndolo en una serie de valores numéricos que caracterizan los sonidos vocales de la señal.

A continuación el motor de búsqueda especializado captura la salida procedente del reconocimiento de voz y realiza las búsquedas en tres bases de datos: un modelo acústico, uno léxico y un modelo de lenguaje.

El modelo acústico representa los sonidos acústicos de una lengua, y puede ser entrenado para reconocer las características de los patrones del habla de un usuario en particular y entornos acústicos.

El léxico enumera un gran número de las palabras en la lengua, y proporciona información sobre cómo se pronuncia cada palabra.

El modelo de lenguaje representa la forma en que se combinan las palabras de un idioma.

- **DRAGON NATURALLY SPEAKING**

Es de licencia propietaria, concretamente de Nuance. Escrito en C++/C [46] [47].

El precio de una versión de estudiantes sería 75 $(54.8€), en cambio la versión premium valdría 175 $(128€).

Funcionalidades de *Dragon Naturally Speaking*:

Los desarrolladores pueden crear rápida y fácilmente aplicaciones de voz o añadir el reconocimiento de voz a las aplicaciones existentes. El motor de voz Dragon Naturally Speaking permite a los usuarios finales acceder, editar y corregir texto por voz, así como el dominio y control de la aplicación mediante la voz.

Logra tasas de exactitud de hasta el 99% además de permitir el dictado continuo en una ventana.

Se han incluido nuevas mejoras como por ejemplo la incorporación de nuevos modelos acústicos para una mejor cobertura de los acentos no nativos y regionales [46].

Algunas de las características de este software de reconocimiento son:

Vocabulario

El vocabulario del *Dragon Naturally speaking* consiste en un modelo de lenguaje y archivos que contienen las palabras de la forma en la que los usuarios suelen pronunciarlas. Los desarrolladores que usen *Dragon Naturally speaking* pueden crear vocabularios personalizados según sus intereses.

Dictado

Los desarrolladores pueden también aprovechar el conjunto completo de características de este software como un dictado de texto de hasta 160 palabras por minuto; corrección or voz para la operación de manos libres, y la reproducción de dictado.

Los comandos de voz

Los desarrolladores pueden codificar sus aplicaciones para activarse cuando detecten una secuencia especificada por código o un patrón de palabras definido con anterioridad y que debe ser reconocido mediante un micrófono.

Texto a voz

Las aplicaciones pueden transformar un texto en una voz sintetizada de salida.

Integra transcripciones de archivos de audio

Se pueden transcribir archivos de audio en archivos de texto y documentos.

- **CMU SPHINX**

Se trata un grupo de sistemas de reconocimiento de voz desarrollados por la Universidad de Carnegie Mellon. Incluye programas para reconocimiento de voz (Sphinx 2 - 4) y un entrenador de modelo acústico (SphinxTrain) [48].

Es de licencia libre BSD y la versión 4 está escrita en java.

Sphinx dispone de gran cantidad de información para desarrolladores.Hay diferentes versiones de este software:

Sphinx 1:

Sphinx es un sistema de reconocimiento de habla continua de un amplio vocabulario, utiliza Modelos ocultos de Márkov (HMMs) y un lenguaje de modelado estadístico de n-gramas. Fue desarrollado por Kai Fu-Lee.

Sphinx 2:

Desarrollado originalmente por Xuedong Huang en la Universidad Carnegie Mellon.

Incorpora funciones tales como, end-pointing, partial hypothesis generation y dynamic language model switching. Es utilizado en sistemas de diálogo y los sistemas de aprendizaje de idiomas.

Sphinx 3:

Sphinx 3 está en desarrollo y proporciona acceso a una serie de técnicas de modelado modernas, como LDA / MLLT, MLLR y VTLN, que mejoran la precisión en el reconocimiento. El objetivo de esta actualización es conseguir un reconocimiento en tiempo real para usar en aplicaciones interactivas.

Sphinx 4:

Es una completa re-escritura de la máquina de Sphinx, con el objetivo de proporcionar un marco más flexible para la investigación en reconocimiento de voz, está escrito íntegramente en lenguaje de programación Java. Sun Microsystems apoya el desarrollo de la Sphinx 4 y contribuyó en el proyecto. Entre los participantes había personas pertenecientes al MIT y la CMU (Carnegie Mellon University).

- **Julius**

Se trata de un software de reconocimiento de voz de amplio vocabulario y de código abierto [49] [50].

Algunas otras características de la librería Julius son:

- Estado de desarrollo: 4 - Beta, 5 - Production/Stable.
- Público destinatario: desarrolladores o usuarios finales.
- Licencia: OSI aprobada, otra/licencia propietaria
- Lenguaje Natural: Inglés, Japonés.
- Sistema Operativo: Linux, Windows, OS Independent
- Lenguaje de Programación: C
- Interfaz de usuario: consola

- **HTK (Hidden Markov Model Toolkit)**

HTK fue desarrollado en el laboratorio de inteligencia artificial de la universidad de Cambridge [51].

HTK es un conjunto de herramientas para la construcción y manipulación de modelos ocultos de Markov. HTK se utiliza principalmente para la investigación de reconocimiento de voz aunque se ha utilizado en numerosas aplicaciones, incluyendo la investigación de síntesis de voz, reconocimiento de caracteres y la secuenciación del ADN.

HTK está en uso en cientos de proyectos diversos. Consiste en un conjunto de módulos y herramientas disponibles en forma de código fuente C. Las herramientas y módulos proporcionan diferentes formas para el análisis de la pronunciación oral, capacitación del modelo HMM (Hidden Markov Model), pruebas y análisis de resultados.

A pesar de que Microsoft tiene la licencia de propiedad del código original de HTK, cualquier persona puede modificarlo para incluir mejoras.

- **Voce**

Es un conjunto de librerías para reconocimiento de voz, accesibles desde java o C++, que internamente se basan en el uso de Sphinx4 y FreeTTS [52].

Son de licencia libre y poseen una API simple.

FreeTTS es un sintetizador de voz digital open source, escrito completamente en el lenguaje de programación Java que apoya el reconocimiento de voz.

- **Librerías AT&T**

La empresa de telecomunicaciones estadounidense proporciona un SDK (Software Development Kit) para permitir el reconocimiento de voz usando el ecosistema de desarrollo de videojuegos Unity explicado anteriormente. Estas librerías están escritas en C# y la principal desventaja es que es necesario el pago de una licencia para su uso [53].

Son unas librerías que permiten tanto el reconocimiento de palabras individuales como frases, en los diferentes idiomas que tiene disponibles, un total de 19 entre los que se encuentra Español, Inglés, Portugués, Ruso, Ucraniano, Italiano, Francés…

- **Asset WordDetection**

Se trata de una herramienta que sólo se podrá usar con un motor de desarrollo de videojuegos específico, ese motor es Unity. La forma de adquirirla es mediante la tienda online, Unity Store.

Un asset es un archivo que contiene una o varios paquetes, y que podemos importar a nuestro proyecto o exportarlo para poder usarlo en otros trabajos. El asset *WordDetection* se compone de un conjunto de archivos que una vez importados a nuestro proyecto aportan al mismo la funcionalidad de reconocimiento de voz. Entre las características más importantes de esta herramienta se encuentran:

- El lenguaje de programación de desarrollo C#.
- Posibilidad de modificación del código inicial para integración completa en los diferentes proyectos.
- Bajo coste de adquisición, unos 30 $.

4. DESCRIPCIÓN DE LA SOLUCIÓN

4.1. Elección del entorno de programación

Para realizar la elección de la herramienta de desarrollo de videojuegos nos basamos en las siguientes razones:

- Compatibilidad con un lenguaje de programación que actualmente esté en uso, y que se actualice continuamente.
- Abundante información y tutoriales disponibles en la web.
- Preferencia de licencia gratuita.
- Compatibilidad con el máximo número de plataformas posible.
- Uso de gráficos de calidad.

En base a los criterios anteriores y comparando todas las opciones disponibles se realiza la siguiente tabla de correspondencias (tabla 1).

Finalmente se extrae un posible ecosistema de desarrollo de juegos:

Unity, por la gran cantidad de información disponible, por el elevado número de desarrolladores que lo utilizan, por su calidad de gráficos, por la gran cantidad de plataformas que permite y un excelente sistema de animación integrado.

Nombre	Lenguaje	Licencia	Uso	Aprendizaje	Eficiencia	2D /3D	Multi-plataforma
SDL	C	LGPL	En auge	Difícil	Normal	SI/ SI	SI
SFML	C++	LGPL	En auge	-	Normal	SI/ NO	SI
PYGAME	Python	LGPL	En uso	Sencillo	Baja	SI/ NO	SI
ANDENGINE	JAVA	LGPL	Limitado	Poca información	Normal	SI/ NO	NO
LIBGDX	JAVA	LIBRE	En uso	Sencillo	Normal	SI/ NO	SI
XNA	C #	EULA	En auge	Complejo	Normal	SI/ SI	SI
UNITY	C#/ C++	Versión gratuita ó pago	Masivo	Sencillo Mucha información	Normal	SI/ SI	SI
OGRE3D	C++	LIBRE	Limitado	Intermedio	Normal	SI/ NO	SI
MARMALADE	Html5/ C++	149$	En uso	Sencillo	Normal	SI/ SI	SI

Tabla 1. Comparativa de frameworks y entornos de programación.

4.2. Elección de la librería de reconocimiento

A la vista de las diferentes librerías de reconocimiento de voz descritas, la que mejor se adapta a las necesidades del proyecto y la que mejores prestaciones ofrece es SPEECH SDK 5.1, por los siguientes motivos:

- Escrita en C++/C#, con lo cual es compatible con el lenguaje de programación elegido en el punto anterior.
- Licencia gratuita.
- Gran cantidad de información disponible.
- Librerías disponibles para la plataforma de programación Microsoft Visual Studio.
- Elevada calidad en el reconocimiento.

La segunda opción que se manejaría como para implementar el reconocimiento de voz, serían las librerías proporcionadas por AT&T, por su excelente acierto al reconocer palabras y por estar escritas en C#.

Finalmente se eligió como posible tercera opción, el asset *WordDetection*. Y se puso en tercer lugar porque la calidad del reconocimiento ofrecida originalmente no llegaba al nivel de Speech SDK ni de las librerías de AT&T. La principal ventaja de *WordDetection* es su fácil integración y su completa compatibilidad con el motor de desarrollo Unity.

En la tabla 2 se muestra la comparativa entre librerías de reconocimiento:

Nombre	Lenguaje	Licencia	Uso	Información	Eficiencia
SPEECH SDK 5.1	C++/C#	Libre	En uso	Abundante	Buena
DRAGON NATURALLY SPEAKING	C/C++	Nuance	En uso	Normal	Muy buena
CMU SPHINX	Java	Libre(BSD)	En uso	Normal	Aceptable
JULIUS	C	Libre (BSD)	En uso	Normal	Aceptable
HTK	C	Gratuito	Medio	Poca	Regular
VOCE	Java/ C++	Libre	Medio	Regular	Aceptable
LIBRERÍAS AT&T	C#	Pago	En uso	Normal	Excelente
WORDDETECTION	C#	Pago	En uso	Normal	Regular

Tabla 2. Comparativa de librerías de reconocimiento de voz.

4.3. Descripción de problemas y soluciones propuestas

Una vez que se tiene elegida la librería de reconocimiento se procede a probarla en Unity para comprobar su compatibilidad.

La librería Speech de Microsoft se encuentra implementada dentro de la librería del sistema de Windows (System) desde la versión de Windows Vista. En pruebas que se realizaron con Visual Studio, la IDE (Integrated Development Environment) proporcionada por Microsoft, se pudo comprobar que dicha librería tenía una fiabilidad de reconocimiento medio-alta.

A continuación se probó la librería de reconocimiento en la IDE proporcionada por Unity, llamada Monodevelop, y no se obtuvieron los resultados esperados. Resultó que Unity no soporta actualmente esta librería, porque su uso se restringe a sistemas operativos de Windows, y Unity es multiplataforma.

4.3.1. Alternativa librerías AT&T:

Como alternativa al fracaso de integración de la librería Microsoft Speech al ecosistema Unity, se encuentra el SDK de reconocimiento de AT&T. Este conjunto de librerías tiene una opción específica para usarse en Unity, por eso en este caso no se tendrían problemas de compatibilidades.

Después de realizar las configuraciones y modificaciones necesarias para el buen funcionamiento de estas librerías, se comprobó que la fiabilidad de las mismas era excelente. Sin embargo, el retraso existente entre la pronunciación de las palabras y la respuesta de reconocimiento obtenida era demasiado elevado. La existencia de este retraso era debida a que se enviaba el paquete de datos con las palabras a reconocer hasta los servidores de la empresa AT&T, y estos servidores enviaban una respuesta que contenía las palabras reconocidas. Entonces se intentó buscar otra solución más rápida para el reconocimiento.

4.3.2. Alternativa asset "WordDetection":

Después de decidir que el uso de las librerías de AT&T no se adaptaba a nuestras necesidades, se pasó a usar la tercera opción un conjunto de librerías desarrolladas por la empresa TheyLoveGames® cuyo objetivo era reconocer palabras individuales, por tanto encajaban perfectamente en la funcionalidad que se quería integrar al videojuego [54]. Después de conocer que dichas librerías estaban escritas en C# y que se podía modificar su código libremente se decidió comprarlas para usarlas en este proyecto. Además el proceso de reconocimiento no necesita de una conexión a internet para su funcionamiento. El nombre dado a estas librerías es *WordDetection*.

Una vez adquiridas, se pudieron realizar unas cuantas pruebas de fiabilidad y aun teniendo una fiabilidad menor que las librerías de AT&T, el retraso en el reconocimiento era inapreciable. Por tanto sopesando las ventajas y desventajas, la solución finalmente elegida para integrar el reconocimiento de voz en nuestro juego fue el asset de Unity llamado *WordDetection*. Sabiendo que el término asset se refiere a un conjunto de recursos que pueden exportarse o importarse a la herramienta Unity [54].

5. DISEÑO DEL VIDEOJUEGO

Se pretende determinar las características tanto funcionales como gráficas que tendrá el videojuego.

Primero se elegirán los elementos de juego que ayudarán a este proyecto a ser enmarcado dentro del término juego serio. Con estos elementos el proyecto será capaz de motivar al usuario en el proceso de aprendizaje y de guiarle para que continúe realizando pruebas y ejercicios mientras aprende.

Después se hará una descripción general funcional del videojuego, en la que se explicará cómo es la organización de la estructura interna del proyecto. Esta parte vendrá acompañada de un diagrama de flujo.

Finalmente se explicará el funcionamiento interno del videojuego desde un nivel más inferior, describiendo las clases y los paquetes necesarios para la ejecución de este proyecto.

5.1. Elementos “gamification”

Un primer paso a realizar es la elección de los elementos más idóneos usados en los juegos serios que encajan en la visión del presente proyecto. En apartados anteriores se explicaron dichos elementos y en base a su funcionalidad y a la motivación que provocaba cada uno de ellos se eligieron los siguientes para ser integrados en nuestro videojuego:

- Marcador de puntuación:

  Si el reconocimiento de la palabra es correcto subirá la puntuación del jugador y se activarán algunos premios por el escenario. Si por el contrario es erróneo disminuirá la puntuación y no aparecerán dichos premios. La puntuación más baja posible será 0. La puntuación ayuda al jugador a llevar un control de lo bien o de lo mal que está realizando las tareas, en definitiva es un feedback usuario-juego.

- Insignias o recompensas:

  Estarán representadas por los premios que aparecerán en el escenario tras superar con éxito el juego de reconocimiento. La existencia de recompensas en el videojuego ayuda a mantener la motivación del usuario.

- Avatares:

  Posibilidad de elegir uno de los dos personajes disponibles en la pantalla de inicio del videojuego: un niño pequeño y un extraterrestre. Este elemento ayuda a mejorar la implicación completa del usuario en el videojuego y a despertar un mayor interés por el mismo.

- Historia:

  El videojuego tiene una historia intrínseca que lo hace más entretenido y atractivo hacia el usuario. Se trata de las aventuras de un niño pequeño que se desvela durante la noche y se encuentra con un monstruo. El niño se ve atrapado por la mente del personaje y para salir de ella tiene que susurrarle una determinada palabra en inglés que será la mostrada por pantalla mediante un dibujo representativo.
  La existencia de una historia completa hace aún más atractivo el videojuego.

El videojuego propuesto y descrito en este proyecto será integrado en una aventura gráfica en la que el jugador tendrá libertad de movimiento y podrá moverse por el escenario de la manera que él estime. Será capaz de saltar entre las plataformas disponibles para conseguir premios o se podrá limitar a realizar las pruebas distribuidas en las diferentes partes de los escenarios. Entre esas pruebas se encontraría la relacionada con el aprendizaje en la pronunciación en inglés, que es el videojuego de menor envergadura desarrollado en este proyecto.

La libertad de jugabilidad que ofrece la aventura gráfica contribuirá en gran medida a la motivación del usuario, elemento importante en un juego serio.

5.2. Diagrama de flujo

En esta sección se pretende explicar cómo funciona el videojuego desde un punto de vista general, para tener una concepción del mismo global y así posteriormente poder reflejar con detalle cada parte integradora del videojuego.

En la figura 1 podemos observar como el videojuego arrancaría en la escena principal llamada *IntroGame*. A partir de ella podríamos elegir distintas opciones: Salir de la aplicación, mostrar la pantalla de ayuda o seleccionar un personaje para empezar a jugar.

Si se elige y se selecciona la opción *Ayuda* se mostraría una pantalla extra que reflejaría los principales controles del videojuego. Se saldría de esta pantalla hacia la pantalla principal pulsando la tecla *b.* Por otro lado si se decide empezar a jugar, primero hay que elegir un avatar de los disponibles, y a continuación ya se pasaría a la siguiente escena (la escena 2 llamada *DreamsMonster).* Una vez en esta escena, se puede circular por el escenario de forma libre. Sin embargo si se toca al monstruo de los sueños, se podría pasar a la escena 3 llamada *VoiceRecognition.* Esta escena es la que realmente ayuda al usuario a mejorar la destreza de la pronunciación oral en la lengua inglesa. En esta última escena se tendrá que pronunciar una palabra especifica, si el reconocedor estima que la pronunciación ha sido la adecuada se subirá el marcador del personaje y se cambiará la palabra que tendrá que ser reconocida en la siguiente ocasión que el usuario entre en esta escena. En caso contrario, si la pronunciación es estimada incorrecta por el reconocedor, se disminuirá la puntuación del usuario y si además se ha intentado con anterioridad pronunciar esa palabra un total de 3 veces y las 3 han resultado erróneas se procederá a cambiar la palabra a reconocer. Si el número de intentos a reconocer es inferior a 3, se mantendrá la misma palabra para ser reconocida. Se da un margen de intentos al usuario para obligarle, hasta cierto punto a intentar pronunciar bien la palabra seleccionada y no evitar la primera vez la pronunciación de la palabra requerida.

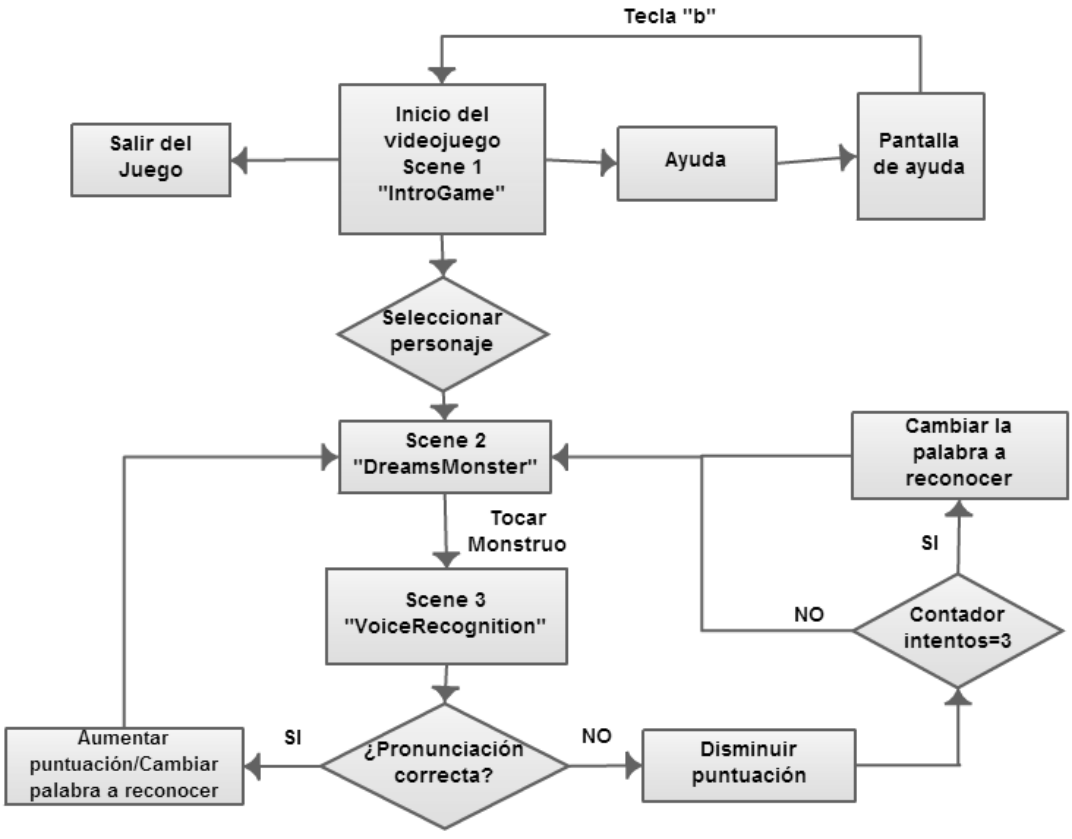

Figura 1. Diagrama de flujo del videojuego.

5.3. Diagramas UML

En este apartado se pretende definir el diseño estructural del videojuego de aprendizaje de la lengua inglesa. Se realizará mediante el uso de los diagramas UML (Unified Modeling Language) [55].

El videojuego, que es lo que llamamos *Proyecto general* se encuentra formado por diferentes paquetes donde cada uno de ellos aporta una determinada característica para el correcto funcionamiento del mismo.

En la figura 2 podemos observar la estructura de nuestro videojuego y las relaciones de los diferentes paquetes que lo forman.

Los paquetes que forman el *Proyecto general* son: *Terrain "e2d"*, que permite la creación y modificación de un terreno 2D que va a ser incorporado al videojuego, el paquete; *WordDetection*, que permite integrar el reconocimiento de voz al proyecto; y finalmente, el paquete *Estructura videojuego* que incluye todos los recursos necesarios para permitir el funcionamiento de los distintos escenarios del videojuego.

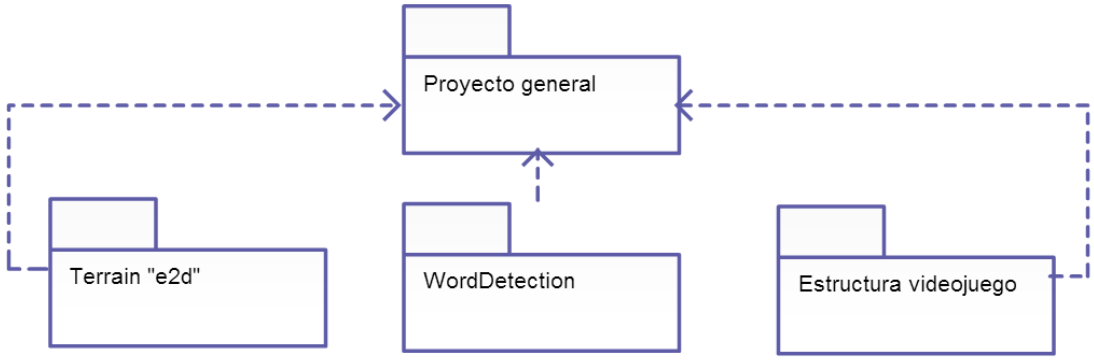

Figura 2. Diagrama UML general del videojuego.

A continuación se procede a explicar el funcionamiento interno de cada uno de los paquetes anteriores.

El paquete *Terrain "e2d"* se descargó como un *asset* de la tienda de Unity. Incluye los recursos necesarios para realizar un terreno de videojuegos en 2D. El funcionamiento de este *asset* consistía en la materialización de un editor gráfico en Unity para la creación de terrenos de forma semiautomática. En la figura 3 podemos ver la organización de subpaquetes que necesita esta herramienta para funcionar:

- Editor: es un paquete que se encarga de las opciones de modificación del terreno, ya sea mediante pincel o agregación de nodos de terreno.

- Generador: se encarga de establecer propiamente el terreno en la pantalla de visualización de Unity. Ya sean terrenos aleatorios con formas establecidas por defecto o creaciones desde cero.
- Recursos: es un conjunto de imágenes, formas y texturas variadas para dar relleno y vistosidad al terreno en cuestión. Se pueden ampliar a gusto del usuario.
- Utilidades: organiza las interacciones que podría tener el terreno con otros elementos incluidos en el mismo escenario de juego.

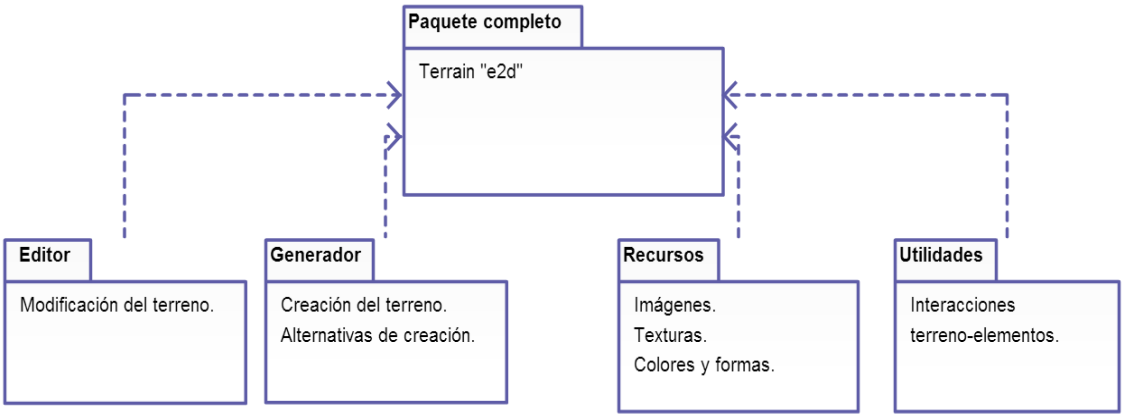

Figura 3. Diagrama UML del gestor de terrenos 2D.

El paquete *WordDetection* que permite el uso del reconocimiento de voz en el videojuego y su contenido se representa en la figura 4. El funcionamiento se basa en el uso de un conjunto de scripts ó archivos de órdenes: *WordDetection*, *WordDetails*, *SpectrumMicrophone* y *FourierTransform*. Como ya se dijo anteriormente están implementados en el lenguaje de programación C# (extensión ".cs").

Para que el reconocimiento de voz funcione, se harán una serie de pasos previos:

a) Se grabarán un conjunto de palabras que más tarde serán las que *WordDetection* podrá reconocer. Este paso se puede realizar de dos formas posibles:

   a.1) Ejecutando los ejemplos de uso que vienen originalmente con el sistema. Los cuales permiten definir la etiqueta de la palabra a grabar, la frecuencia de muestreo y el tiempo de captura.

   a.2) Grabando las palabras necesarias con una herramienta de audio externa. Esta solución se tuvo que realizar adicionalmente porque no estaba disponible originalmente.

b) Se guardarán las palabras en el registro del sistema. Lo que realmente se incluye en el registro son las muestras de las palabras grabadas junto con la etiqueta de la palabra.

c) A continuación se cargarán las palabras del registro al ejecutarse el juego por primera vez.

Una vez se tienen las palabras grabadas ya se puede utilizar el reconocimiento de voz.

A continuación se describen las funciones que realizan los scripts de *WordDetection* representados en la figura 4:

La clase *FourierTransform* se encarga de calcular la Transforma de Fourier de las muestras de las palabras grabadas. De esta forma se calcula el espectro (real e imaginario) de cada una de las palabras disponibles en el registro del sistema.

La clase *SpectrumMicrophone* es la encargada de realizar la captura de las palabras que se quieren grabar en el registro y de las palabras pronunciadas por el jugador del videojuego. Define el método y las características de grabación, como el tiempo de captura y la frecuencia de muestreo.

La clase *WordDetails* es la que guarda y registra las propiedades de cada una de las palabras grabadas, tanto las pertenecientes a la base de datos de referencia como las pronunciadas por el usuario del videojuego. Concretamente la clase tiene como propiedades las muestras de las grabaciones, la etiqueta que identifica a la grabación y las muestras de los espectros real e imaginario calculados con la clase *FourierTransform*.

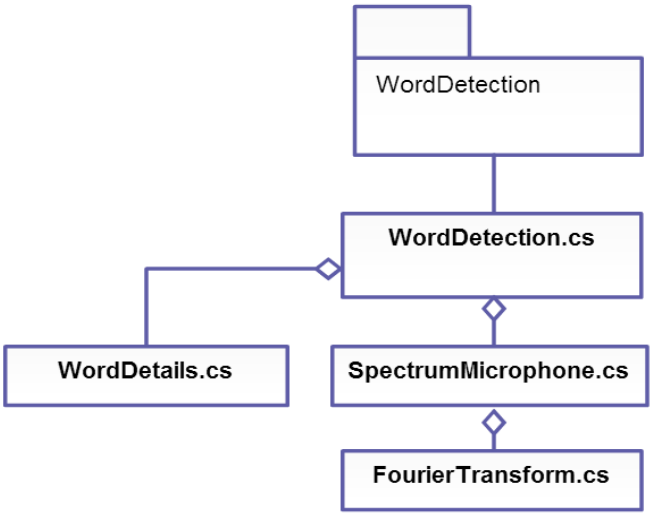

Figura 4. Diagrama UML de WordDetection.

La clase de alto nivel que usa las tres clases descritas anteriormente es *WordDetection.* Esta clase permite cargar y guardar las palabras del registro del sistema, además de calcular la palabra reconocida. El método de elección de la palabra que ha sido pronunciada por el usuario sigue la siguiente estructura:

a) Una vez calculado el espectro de cada una de las palabras de la base de datos de referencia, se recorren las palabras una por una y se compara el espectro de la palabra pronunciada por el usuario con el de las palabras de referencia.
b) Se va calculando en cada comparación la diferencia numérica entre los espectros y se va guardando en una variable.

c) Después de realizar tantas comparaciones y cálculos como palabras existen en el registro, cada una de las palabras del registro tendrá un valor de puntuación que indicará el grado de similitud con la palabra pronunciada por el usuario.
d) La mínima de estas puntuaciones es la palabra reconocida.
e) Finalmente se lanzará un evento que indicará la palabra que se ha reconocido. Concretamente el evento que se genera es una variable de la clase *WordDetails*.

A continuación vamos a describir la composición del paquete *Estructura videojuego* que está relacionada con la distribución de escenarios del videojuego. En la figura 5 se observa mejor esta distribución.

La escena 1 *IntroGame* es la pantalla de inicio del videojuego, en la que se toman las elecciones oportunas antes de empezar a jugar.

La escena 2 *DreamsMonster* es el escenario de juego principal.

Y la escena 3 *VoiceRecognition* que es donde se realiza el reconocimiento de voz.

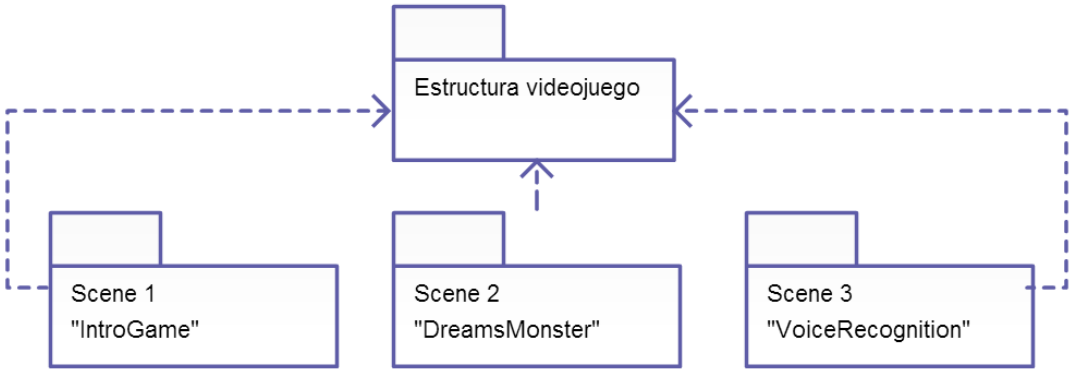

Figura 5. Diagrama UML de estructura de escenas del videojuego.

Asociados a la escena 1 se encuentran dos scripts (figura 6). El primero de ellos, *IntroGame.cs* gestiona los procesos asociados a los 3 botones del videojuego de la escena 1. Uno de estos botones permite la selección del personaje principal, otro la aparición de la ventana de ayuda con indicación de los controles de juego y el restante la detención del videojuego.

El segundo llamado *Recognition.cs* se ocupa de cargar las palabras existentes en el sistema para tenerlas en memoria y así permitir el funcionamiento de *WordDetection* en la escena 3. En la figura 6 se observan con claridad.

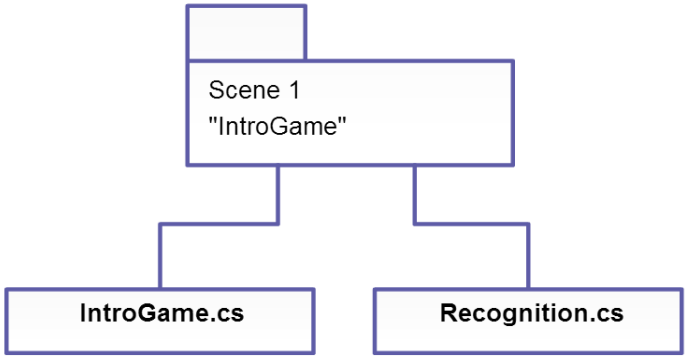

Figura 6. Diagrama UML de escena 1.

Una vez se ha elegido un personaje se pasa al escenario siguiente llamado *DreamsMonster*. En este escenario nuevo entran a funcionar diferentes scripts, como vemos en la figura 7.

- Los scripts asociados al personaje principal son dos:

CharcaterAnimator.cs: permite el cambio continuo de un conjunto de imágenes que muestran al personaje en diferentes posiciones de movimiento, ya sea saltando o andando. Este cambio de imágenes se produce cada cierto tiempo y es lo que permitirá distinguir al personaje en movimiento.

RigidBodyFPSController.cs: permite controlar las órdenes que se introducen mediante el teclado para mover al personaje. Los movimientos gestionados son el salto hacia la izquierda y hacia la derecha, andar hacia la izquierda y hacia la derecha y agacharse.

- Por otro lado tenemos un script asociado al monstruo que aparece en escena:

LoadScene2.cs: se encarga de cargar la escena 3 cuando el personaje principal toca al monstruo.

- También existe un script asociado al comportamiento de los puntos extra del videojuego, *Coin.cs* . Su funcionalidad es la de incrementar o decrementar el marcador de puntuación del usuario. Además permite visualizar monedas en el escenario cuando el usuario realiza correctamente las pruebas de reconocimiento del videojuego, y las hace desaparecer cuando el personaje entra en contacto con las mismas.

- Finalmente se usó un último script que es el que se encarga de llevar la cuenta del número de intentos que se llevaron a cabo en la pantalla de reconocimiento de voz.

Es el script llamado *TimesRecognition.cs*.

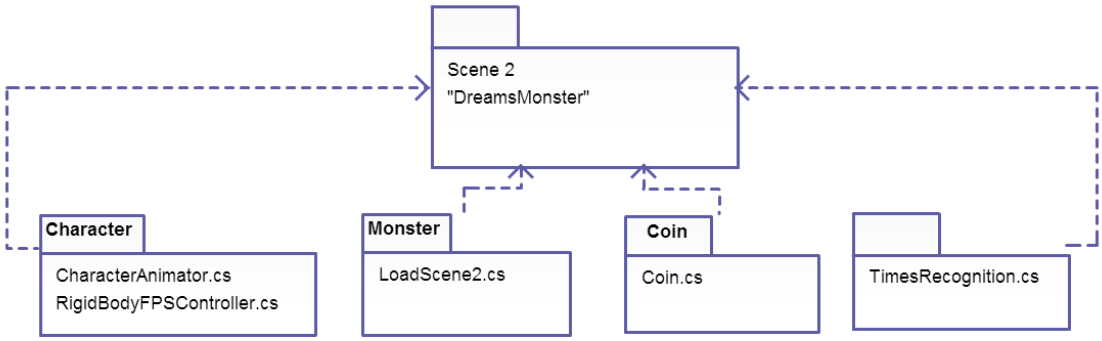

Figura 7. Diagrama UML de escena 2.

Para el correcto funcionamiento de la escena 3 se necesitan un conjunto de scripts que son explicados a continuación.

En la figura 8, vemos como es necesario asociar al personaje principal el script *AnimateEnable.cs* que es el que desplazará al personaje de forma automática en varias ocasiones: cuando entre a la pantalla principal, y después de haber pronunciado la palabra a reconocer.

Por otro lado se encuentra el script *ExampleOneWord.cs* que es el que llama a las funciones del asset *WordDetection* para capturar la palabra que ha pronunciado el usuario en esta escena. Después de muestrear la palabra, calcular el espectro y obtener la palabra de la base de datos que más se parece a ésta, se mostrará el resultado y se informará al usuario de su éxito o fracaso. Finalmente el script regresará al escenario anterior cambiando la puntuación del usuario en base a sus resultados de reconocimiento.

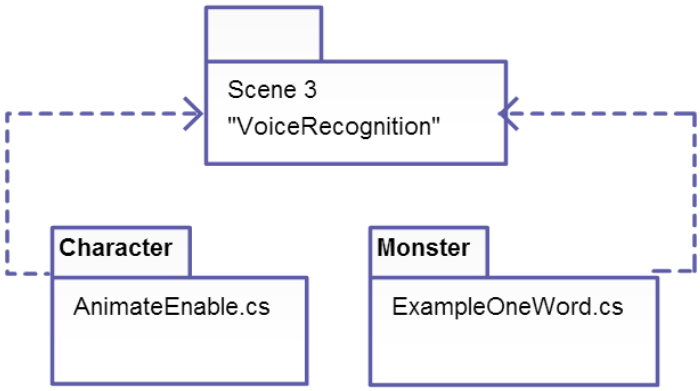

Figura 8. Diagrama UML de escena 3.

5.4. Matriz de dependencias

Una vez que se ha explicado el funcionamiento de las clases que forman nuestro videojuego, se pretende realizar una matriz de dependencias en la que se observe visualmente las clases junto a las tareas que realizan cada una de ellas.

En otras palabras se trata de una matriz que refleja las funciones que se deben cubrir en el videojuego junto a las clases que se han tenido que implementar para llevarlas a cabo. La organización de las funciones se puede visualizar en la figura 1.

En la tabla 3, se observa cómo se ha dividido el proyecto general en funciones y cómo cada una de ellas está resuelta con una o varias clases:

- Inicio del videojuego:
  Se encarga de cargar la escena 1 del videojuego, en la cual se puede elegir un personaje, seleccionar la opción de ayuda o cerrar el programa.

- Ayuda y pantalla de ayuda:
  Permite gestionar la visualización de la pantalla de ayuda desde la escena 1 de inicio del videojuego, dependiendo de la elección del usuario.

- Salir del juego:
  Gestiona el cierre de la aplicación cuando el usuario lo seleccione.

- Seleccionar personaje:
  Posibilita la elección de uno de los dos personajes para el juego.

- Escena1-Escena2:
  Es el traspaso de la escena 1 a la escena 2 del videojuego una vez se ha elegido al personaje con el que se desea jugar.

- Fondo escena 2:
  Permite gestionar y modificar el fondo del juego durante toda la escena 2 del mismo.

- Escena2-Escena3:
  Gestiona el paso de la escena 2 a la escena 3, cuando el personaje entra en contacto con el monstruo situado en la escena 2.

- Reconocimiento de voz:
  Se encargar del funcionamiento de la escena de reconocimiento de voz. Determina si la pronunciación ha sido correcta/incorrecta y en base a ello ejecuta la lógica del contador de intentos de pronunciación definida en la figura 1. Además también se encarga de la gestión del cambio de la palabra que se requiere que sea reconocida.

- Marcador/Puntuación:

Gestiona el incremento o decremento del marcador en base a los resultados positivos/negativos obtenidos en la escena 3 de reconocimiento de voz ("VoiceRecognition").

Como funciones generales presentes durante el funcionamiento del videojuego están:

- Movimiento del personaje:
  Se encarga del desplazamiento del personaje por la pantalla con los controles determinados.

- Animación movimiento:
  Consiste en la actualización de un conjunto de texturas con diferentes posiciones del personaje a una cierta velocidad y con un contador de tiempo adecuado a la velocidad de movimiento del personaje.

- Pausar el juego:
  Consiste en parar/activar la ejecución del videojuego cuando el usuario así lo determine.

- Gráficos:
  Gestiona el cambio de imágenes que se están visualizando en las escenas en base al funcionamiento del videojuego.

CLASES	Inicio videojuego	Ayuda/Pantalla ayuda	Salir juego	Seleccionar personaje	Escena1-escena2	Fondo escena 2	Escena2-escena3	Reconocimiento Voz	Marcador/Puntuación	Movimiento personaje	Animación movimiento	Pausar juego	Gráficos
	ESTRUCTURA VIDEOJUEGO												
IntroGame	X	X	X	X	X								X
Recognition								X					
CharacterAnimation											X		X
RigidBodyFPSController										X			
Coin									X				X
TimesRecognition							X	X	X				
AnimateEnable									X		X	X	X
LoadScene2							X						X
ExampleOneWord								X					
CameraGradienteBackground						X							X
Help		X											X
	WORDDETECTION												
WordDetection								X					
WordDetails								X					
SpectrumMicrophone								X					
FourierTransform								X					
	TERRAIN "E2D"												
e2dCurveMethods													X
e2dMidpoint													X
e2dPerlinNoise													X
e2dPerlinOctave													X
e2dVoronoi													X

Tabla 3. Matriz de dependencias.

5.5. Entorno de sistema

A continuación se van a describir las características del sistema que se usó para la realización del proyecto.

Como el videojuego está desarrollado con el motor gráfico Unity®, para el correcto funcionamiento de dicha herramienta son necesarios los siguientes requisitos mínimos [56]:

- Versiones sistema operativo Windows: Windows XP con SP2 o posterior; Windows 7 con SP1 o posterior; Windows 8; No soporta Windows Vista y no se probó en las versiones de servidor de Windows.
- Versiones sistema operativo Mac: Mac OS X Snow Leopard 10.6 o posterior. No se probó en las versiones de servidor de OS X.
- Tarjetas gráficas: Tarjeta gráfica con capacidad para DirectX 9 (shader modelo 2.0).

El sistema final usado consta de las siguientes características:

- Modelo PC: HP.
- Sistema operativo: Windows 7 Home Premium 64 bits.
- Procesador: Intel® Core™2 Duo @ 2,26 GHz.
- RAM: 4GB

Tarjeta gráfica con capacidad para DirectX 1

6. RESULTADOS DEL RECONOCIMIENTO

Se pretende estudiar la fiabilidad del reconocimiento de voz a través del asset de Unity llamado *WordDetection*. Para ello se han realizado una serie de pruebas que evalúan su calidad. A partir de los resultados de las pruebas se tomará la decisión final sobre qué frecuencia de muestreo es la idónea, qué tiempo de captura proporciona mejores resultados o cuántos locutores distintos es conveniente que existan en la base de datos de referencia.

El correcto funcionamiento de *WordDetection* se basa en un conjunto de grabaciones de palabras que previamente se han realizado y se han guardado en el registro del sistema. Lo que realmente se guarda en el registro es la etiqueta y las muestras de cada palabra grabada.

Hay que destacar que en el registro del sistema además de las palabras grabadas también tiene que estar incluida una opción extra que es el ruido de ambiente capturado.

Una vez que se tiene en el registro la base de datos de palabras que se quiere que sean reconocidas, se puede usar el reconocedor. Se pronunciará una palabra que se sabe que está previamente guardada, y a continuación el asset se encargará de muestrear la señal asociada a la palabra. Se calculará la Transformada de Fourier de estas muestras y se comparará este espectro con el espectro de cada una de las palabras de la base de datos. En base a dicha comparación, se genera una puntuación, habrá tantas puntuaciones como palabras haya en la base de datos. La mínima de estas corresponderá con la palabra que más se asemeja a la palabra pronunciada al micrófono, y será por tanto la palabra reconocida.

Para poder llevar un control de las puntuaciones obtenidas, se creó un archivo de texto en el que se guardaban todas las puntuaciones asociadas a las grabaciones cada vez que se pronunciaba al reconocedor una determinada palabra.

Cabe destacar que originalmente *WordDetection* mantenía el micrófono activo durante todo el tiempo que el reconocedor estaba funcionando. Pero se realizaron algunas modificaciones para que después de pronunciar una palabra se detuviese el reconocedor, y así analizar las puntuaciones obtenidas en cada pronunciación de una manera inequívoca.

Además también se tuvo que modificar la forma de grabación del asset de reconocimiento. Inicialmente se grababan las palabras de la base de datos de una forma poco precisa, pues había que pulsar un botón hasta que acabases de hablar, lo cual llevaba a tener tamaños de grabaciones

distintos. Por eso se realizaron las modificaciones necesarias en el asset para cargar archivos *.wav* con las palabras grabadas y así muestrearlos y guardar los resultados del muestreo en el registro.

Teniendo una base de datos de palabras grabadas en formato *.wav* se consiguen varias ventajas tales como:

- Se evita el inconveniente de que se borren los datos del registro por cualquier motivo y desaparezcan las palabras.
- Se tiene un mayor control de las señales y el ruido capturado en cada una de las grabaciones asociadas a las palabras, pudiendo reducir el ruido o acortar la grabación.

Una vez se tiene una visión generalizada del funcionamiento del reconocedor, se procede a describir los tipos de pruebas realizadas y las conclusiones que se extraen en cada una de ellas.

6.1. Grabación de palabras con un único locutor

Se graban en el registro del sistema dos palabras, *jeans* y *house*. En esta prueba el locutor elegido para realizar las grabaciones es femenino.

Una vez se tienen las palabras en el registro, se activa el reconocedor y el mismo locutor que hizo las grabaciones pronuncia un número de 20 veces seguidas cada una de las palabras existentes en el registro. En este caso particular, 20 veces *jeans* y 20 veces *house*. A continuación se guardan las puntuaciones que el reconocedor ha ido calculando para las palabras en cada una de las 40 veces en las que ha intervenido el locutor.

El procedimiento anterior se realiza para diferentes configuraciones de frecuencia de muestreo y tiempo de captura. Las combinaciones de configuración elegidas se muestran en la tabla 4:

TIEMPO DE CAPTURA	FRECUENCIA DE MUESTREO
1 segundo y 2 segundos	8192
1 segundo y 2 segundos	16384
1 segundo	32768

Tabla 4. Opciones tiempo de captura y frecuencia de muestreo.

Finalmente se transcriben los resultados de puntuación calculados por *WordDetection* de todos los casos descritos a una hoja de cálculo y se realiza un análisis estadístico de los mismos. Este análisis consiste en calcular la relación entre la puntuación de la palabra que debe ser reconocida y la puntuación de la palabra sobrante. Como hay dos palabras a reconocer hay que realizar este análisis dos veces por cada combinación de tiempo de captura y frecuencia de muestreo. Obtendremos al final un total de diez relaciones.

La columna *jeans* corresponde a la relación calculada cuando se pronunciaba *jeans* al micrófono y la columna *house* cuando se pronunciaba *house* al micrófono. Por tanto cuanto menores sean estas

relaciones, mayor margen habrá entre la palabra correcta y la errónea. Las relaciones calculadas para diferentes configuraciones se muestran en la tabla 5:

	JEANS	HOUSE
Tiempo captura: 1seg. Frec: 8192	0,24267933	0,71410017
Tiempo captura: 1seg. Frec: 16384	0,514192921	0,64848881
Tiempo captura: 1seg. Frec: 32768	0,470683795	0,656342827
Tiempo captura: 2seg. Frec: 8192	0,34315299	0,75923303
Tiempo captura: 2seg. Frec: 16384	0,44223485	0,56293274

Tabla 5. Relaciones de *jeans* y *house* con diferentes configuraciones.

Como podemos observar en las tablas, los mejores resultados obtenidos para un tiempo de captura de 1 segundo son los proporcionados por la frecuencia de muestreo de 32768 muestras por segundo. Porque, en un estudio global, reconoce con una calidad similar ambas palabras. Por su parte la frecuencia de 8192 detecta *jeans* con la mayor fiabilidad, sin embargo en el caso de *house* la fiabilidad disminuye considerablemente. Entre la frecuencia de 16384 y 32768 no hay grandes diferencias, pero se ha escogido una frecuencia mayor porque se cogen más muestras en el mismo tiempo de captura y por lógica el reconocimiento tiene que mejorar.

En cuanto a las combinaciones que utilizan un tiempo de captura de 2 segundos. Se observa que la frecuencia de 16384 es mucho mejor que la de 8192.

La frecuencia de 8192 presenta además un problema que la hace inutilizable. Tanto en el caso de que el tiempo de captura de las palabras sea de 1s como de 2s las puntuaciones proporcionadas por *WordDetection* son incoherentes. En la tabla 6 y en la tabla 7 pueden apreciarse dichas incoherencias de puntuación. En dichas tablas aparecen reflejadas las medias de las 20 puntuaciones otorgadas para *jeans* y para *house* cuando existían dos casos de estudio, pronunciar *jeans* y pronunciar *house*. Y la relación es el resultado de dividir las puntuaciones de *jeans* y *house*, teniendo en cuenta que el numerador es la puntuación correspondiente a la palabra que se espera reconocer.

Las principales incoherencias se detectan fácilmente, se obtiene un score menor en la puntuación de *house* al pronunciar *jeans* (fallo) que al pronunciar *house* (acierto) cuando tenía que ser al contrario.

Frec: 8192 1s	JEANS	HOUSE	RELACION
Pronunciar jeans	38,2	**157,43**	0,24
Pronunciar house	250,73	179,05	0,71

Tabla 6. Resultados frecuencia de 8192 y tiempo de captura 1s.

La misma incoherencia sucede con en tiempo de captura de 2s y de frecuencia 8192, al pronunciar *jeans (*fallo*)* se obtiene en *house* una puntuación de 427,64 que es menor que la obtenida en *house* al pronunciar *house*(acierto) 574,21.

Frec: 8192 2s	JEANS	HOUSE	RELACION
Pronunciar jeans	146,74	427,64	0,34
Pronunciar house	756,3	574,21	0,76

Tabla 7. Resultados frecuencia 8192 y tiempo de captura 2s.

Por tanto tenemos dos posibles combinaciones de configuración que destacan en cuanto a fiabilidad se refiere:

- Tiempo captura: 1seg. Frecuencia: 32768
- Tiempo captura: 2seg. Frecuencia: 16384

Utilizando dos segundos de captura, se grabaría una considerable cantidad de ruido ambiente y no nos interesa. Por tanto, se ha decidido elegir la primera configuración. Un segundo de tiempo de captura es suficiente para pronunciar cualquier palabra y una frecuencia mayor dará mejores resultados en el reconocimiento.

A partir de esta prueba, se procederá a realizar otras diferentes usando la configuración del reconocedor de frecuencia 32768 y el tiempo de captura 1 segundo.

6.2. Grabación de 4 palabras con 4 locutores

En esta segunda prueba se procede a estudiar el comportamiento del reconocedor frente a locutores que no han sido los que han grabado las palabras de la base de datos. Estos locutores son los que calificaremos como “externos”.

Se empieza eligiendo a dos locutores femeninos, entre ellos se encuentra el locutor que realizó las pruebas anteriores. Después se eligen a otros dos locutores masculinos. En total tenemos 4 locutores que los llamaremos “internos”.

A continuación, cada uno de estos 4 locutores internos tendrá que grabar 4 palabras distintas para formar la base de datos de 16 palabras. En esta ocasión, las palabras elegidas son *jeans, house, boots y socks*.

Como ya se explicó anteriormente, hay que grabar el ruido ambiente como si fuese una palabra más, por tanto tendremos en esta prueba un total de 17 palabras.

Una vez que tenemos las palabras guardadas en el registro del sistema, ya está preparado el reconocedor.

Finalmente, para probar la fiabilidad del reconocedor con esta nueva base de datos se determina que quienes pronunciarán las palabras al micrófono serán un locutor interno femenino (el mismo que realizó las pruebas en el anterior punto) y dos externos, uno femenino y otro masculino. Cada uno de estos locutores repetirá al micrófono un total de 10 veces las palabras grabadas (*jeans, house, socks* y *boots*). Al final, se obtendrán las puntuaciones que el reconocedor ha proporcionado

en las 120 veces que se ha ejecutado, 40 para el locutor externo masculino, 40 para el externo femenino y 40 para el interno femenino.

Se trascribirán los resultados obtenidos anteriormente a una hoja de Excel y se evaluarán.

- **Evaluación del tipo de análisis estadístico a realizar:**

Para la evaluación se pretenden comparar los resultados obtenidos con el mínimo, el máximo, la media y la mediana.

El procedimiento usando el valor estadístico del mínimo sería el siguiente:

Se calcula el mínimo valor de puntuación que se ha asignado a cada grupo de palabras: 4 *jeans*, *4 house*, 4 *socks* y 4 *boots*. Después de obtener estos mínimos, se calculará la relación entre el mínimo de la palabra que debe ser la reconocida y el mínimo de los 12 valores en las tres palabras restantes.

En la tabla 8 se muestran las puntuaciones que ha calculado el reconocedor de voz cuando un locutor interno femenino (femenino 1) ha pronunciado un número de 10 veces la palabra *jeans*, incluyendo en la última fila la media de todas las puntuaciones de cada una de las columnas. Con estas puntuaciones se pretende explicar un ejemplo concreto de la forma de realización del análisis estadístico.

	JEANS				HOUSE				SOCKS				BOOTS			
Frec.32768	FEMENINO 1	FEMENINO 2	MASCULINO 1	MASCULINO 2	FEMENINO 1	FEMENINO 2	MASCULINO 1	MASCULINO 2	FEMENINO 1	FEMENINO 2	MASCULINO 1	MASCULINO 2	FEMENINO 1	FEMENINO 2	MASCULINO 1	MASCULINO 2
JEANS	482,2	1020	751,4	561,9	1161	1402,6	945,2	1163,06	664,7	767,2	626,5	571,4	605,5	620,2	676,5	899,6
	417,4	1028	750,6	535,1	1109	1331,8	956,24	1118,97	540,2	615,5	550,6	490,3	581,3	577,6	595,7	727,8
	371,6	952,5	689,2	532,6	1072	1240,2	844,8	1054,07	535,3	639,1	503,1	487,3	631,3	485,4	553,1	679,7
	356,9	1052	772	563,0	1090	1321,91	963,5	1109,31	518,1	657,8	510,5	512,7	567,3	532,8	560,8	672,4
	472,9	1043	785,5	550,9	1209	1378,68	961,2	1185,03	625,4	655,7	629,4	560,8	685,1	605,4	699,3	707,3
	470,9	1043	760	556,4	1114	1338,93	1001,3	1136,41	614,1	729,9	562,1	539,1	533,1	546,9	594,5	739,1
	420,5	1064	793,9	543,9	1108	1359,8	1015,3	1149,53	566,9	691,6	555,9	498,1	576,7	558,2	574,1	730,7
	334,5	1106	802,4	582,0	1063	1344,32	1027,3	1117,7	503,9	741,8	493,45	461,2	545,5	510,4	486,5	675,6
	285,4	955,2	663,3	550,8	912,3	1203,6	933,6	966,12	481,9	692,5	386,17	401,4	431,1	356,3	345,1	657,7
	319,4	1083	826,7	585,2	1055	1347,4	1007,1	1121,71	451,9	655,0	461,2	480,8	543,3	503,2	642,5	575,6
MEDIA	393,2	1035	759,52	556,2	1089	1326,9	965,5	1112,1	550,2	684,6	527,9	500,3	570,1	529,7	572,8	706,6

Tabla 8. Puntuaciones locutor interno femenino (femenino 1) al pronunciar jeans.

Además de las puntuaciones mostradas en la tabla 8, hay que tener en cuenta que existen tres tablas más como la anterior, cuando se pronuncia *house*, *socks* y *boots*. En total 4 tablas que también estarán presentes en el caso del locutor externo femenino y el locutor externo masculino.

Supongamos que tomamos como base las puntuaciones anteriores, el siguiente paso es calcular el mínimo de puntuación dentro de cada bloque de 4 palabras, un bloque para *jeans*, *house*, *boots* y *socks*. Por ejemplo para la primera fila de la tabla 8, el resultado obtenido al realizar el cálculo anterior serían 4 puntuaciones (*jeans*: 482,2, *house*: 945,2, *socks*: 571,36, *boots*: 605,55) con las cuales se calcularía posteriormente una relación. Esta relación consiste en realizar una división entre dos valores, el numerador sería la puntuación de la palabra que pronunció el locutor en cada intento y el denominador el mínimo de las tres puntuaciones restantes. Un resumen de los valores de los mínimos y relaciones se puede ver en la tabla 9. En la tabla 9a, como la palabra pronunciada ha sido *jeans* el numerador de la relación sería el valor 482,2 y el denominador sería el mínimo de entre las tres puntuaciones restantes, 571,36. Finalmente el valor de la relación es 0,84395127.

SE PRONUNCIA JEANS				
JEANS	**HOUSE**	**SOCKS**	**BOOTS**	**RELACION**
482,2	945,2	**571,36**	605,55	**0,84395127**
417,48	956,24	490,35	577,63	0,85139186
371,6	844,85	487,3	485,36	0,76561727
356,97	963,5	510,5	532,85	0,69925563
472,9	961,2	560,89	605,43	0,84312432
470,96	1001,3	539,05	533,05	0,88351937
420,5	1015,25	498,02	558,28	0,8443436
334,53	1027,03	461,23	486,5	0,72529974
285,46	912,3	386,17	345,03	0,82734835
319,41	1007,11	451,99	503,24	0,70667493
				0,79905263

(a)

SE PRONUNCIA HOUSE				
JEANS	**HOUSE**	**SOCKS**	**BOOTS**	**RELACION**
899,43	639,7	781,23	729,03	0,87746732
829,7	574,24	797,17	755,4	0,76018004
827,2	657,07	787,4	798,5	0,83448057
994,5	738,97	900,5	869,89	0,84949821
983,27	838,69	939,04	889,14	0,94325978
836,4	609,12	744,23	692,93	0,87904983
738,26	628,54	668,06	644,93	0,97458639
1030,5	651,14	956,45	915,7	0,71108442
961,23	808,07	884,55	841,6	0,96015922
758,37	576,72	660,13	653,41	0,88263112
				0,86723969

(b)

SE PRONUNCIA SOCKS				
JEANS	**HOUSE**	**SOCKS**	**BOOTS**	**RELACION**
707,16	801,6	612,4	726,07	0,86599921
579,74	842,2	476,67	647,82	0,82221341
547,66	766,27	473,68	533,44	0,88797241
475,04	666,5	382,28	435,51	0,87777548
458,96	679,75	371,66	420,23	0,88442044
494,95	732,99	409,71	521,99	0,82778058
495,52	813,77	410,7	535,72	0,82882628
537,5	797,2	454,46	580,76	0,84550698
442,3	753,3	380,34	446,29	0,85991409
462,23	745,26	390,53	470,3	0,84488242
				0,85452913

(c)

SE PRONUNCIA BOOTS				
JEANS	**HOUSE**	**SOCKS**	**BOOTS**	**RELACION**
393,7	829,7	385,96	348,41	0,90271013
389,74	792,32	357,45	322,6	0,90250385
383,3	905,02	394,64	294,9	0,76937125
327,97	857,67	364,24	296	0,90252157
450,3	887,37	418,25	336,23	0,80389719
370,34	825,1	339,8	257,14	0,75673926
382,9	838,53	366,8	262,8	0,71646674
299,63	805,8	300,42	198,06	0,66101525
358,7	897,6	409,6	347,6	0,96905492
354,3	816,5	334,26	254,2	0,76048585
				0,8144766

(d)

Tabla 9. Cálculo de relaciones usando el mínimo.

Para el cálculo de las relaciones anteriores se ignora la puntuación otorgada al ruido.

En la tabla 9 también calculamos el promedio de las relaciones en los 4 casos posibles, cuando el locutor pronuncia *jeans* (tabla 9 a) 0,799, *house* (tabla 9 b) 0,867, *socks* (tabla 9 c) 0,855 y *boots* (tabla 9 d) 0,814.

Hay que tener en cuenta que el mismo análisis anterior se llevó a cabo para los tres locutores del estudio. A continuación se procede a realizar un resumen de los resultados estadísticos obtenidos de la forma explicada anteriormente, es decir, con el mínimo.

En cada casilla de la tabla 10 aparecen los promedios de las 10 relaciones de cada una de las 4 palabras cuando probaron el reconocedor los 3 locutores seleccionados. En el caso de la primera fila y la primera columna, el valor 0,799 es el promedio de las 10 relaciones calculadas después de las 10 repeticiones de *jeans* realizadas por el locutor interno femenino (femenino 1). Ese valor indica la calidad del reconocimiento al pronunciarse la palabra *jeans*. Cuanto más pequeño sea éste, mejor será la calidad del reconocimiento. Si por el contrario fuese un valor elevado (mayor que 1) el reconocimiento hubiese sido erróneo.

En la última fila podemos observar el promedio de las puntuaciones que han obtenido los tres locutores al pronunciar cada una de las 4 palabras propuestas.

	MINIMOS			
LOCUTOR	**JEANS**	**HOUSE**	**SOCKS**	**BOOTS**
INT:FEM	**0,799**	0,867	0,855	0,814
EXT:FEM	0,919	0,702	1,024	0,65
EXT:MASC	0,972	0,871	0,904	0,821
MEDIA	**0,897**	**0,813**	**0,928**	**0,7612**

Tabla 10. Mínimos de varios locutores.

Se realiza la misma tabla anterior pero esta vez usando el máximo como método estadístico (tabla 11).Veamos otro ejemplo esta vez usando los valores proporcionados por el máximo. En la primera fila y primera columna de la tabla 11 se ve 1,54, es el promedio de las 10 relaciones calculadas usando el máximo como valor estadístico y teniendo en cuenta que ha sido el locutor femenino 1 el que lo ha obtenido, pronunciando *jeans*. Concretamente el cálculo de las relaciones consistió en una serie de pasos:

- Calcular los máximos de los 4 bloques existentes en la base de datos (máximo *jeans*, máximo *house*, máximo *socks*, máximo *boots*), para cada uno de los locutores elegidos.
  Por ejemplo, nos fijamos en la primera fila de la tabla 8 que se corresponde con las puntuaciones obtenidas en la primera de las 10 pruebas realizadas por el locutor femenino 1. Los máximos obtenidos de cada uno de los bloques son: *jeans* 1020 (primera fila y segunda columna), *house* 1402,6(primera fila y sexta columna), *socks* 767,2(primera fila y décima columna) y *boots* 899,6 (primera fila y decimosexta columna).

- Dividir el valor máximo de *jeans* obtenido (palabra pronunciada) entre el mínimo de los valores máximos existentes (todo esto un total de 10 veces).
  Con los valores del ejemplo anterior, la relación sería 1,33, resultante de dividir 1020 entre 767,2.

- Hacer un promedio de las 10 relaciones para obtener un único valor. El resultado es el que se dijo al inicio de la explicación, 1,54.

	MAXIMOS			
LOCUTOR	**JEANS**	**HOUSE**	**SOCKS**	**BOOTS**
INT:FEM	**1,54**	0,835	0,954	0,954
EXT:FEM	0,94	0,804	0,982	0,967
EXT:MASC	1,121	0,894	0,985	0,915
MEDIA	1,200	0,844	0,974	0,945

Tabla 11. Máximos de varios locutores.

La misma operación se realiza usando la media y la mediana. En la tabla 12 se muestran por palabras los valores medios de calidad de reconocimiento obtenidos por un total de tres locutores cuando se han usado diferentes alternativas estadísticas.

	JEANS	**HOUSE**	**SOCKS**	**BOOTS**
MÍNIMO	**0,897**	0,813	0,928	0,762
MÁXIMO	1,200	0,844	0,974	0,945
MEDIA	1,059	0,842	0,932	0,816
MEDIANA	1,055	0,890	0,923	0,719

Tabla 12. Comparativa mínimo, máximo, media y mediana.

En base a los resultados obtenidos en la tabla 12 se concluye que el mejor valor estadístico para ser utilizado es el mínimo porque en términos generales proporciona los valores más bajos posibles.

A continuación se calcula el porcentaje de mejora o empeoramiento del máximo, la media y la mediana, fijando como referencia los valores obtenidos del mínimo (B). En la tabla 13 se ha restado a los valores del mínimo los valores de máximo, media y media y se ha multiplicado por 100 para obtener un porcentaje. Si el resultado de este porcentaje es positivo, se habrá producido una mejora en la calidad del reconocimiento ya que el margen de error, una vez se ha pronunciado una determinada palabra, es más grande. En caso contrario el resultado será negativo y el margen

de error más pequeño, pudiendo producirse más fácilmente errores en el proceso de reconocimiento.

	MEJORA/EMPEORA (100*(B-A))			
LOCUTOR	**JEANS**	**HOUSE**	**SOCKS**	**BOOTS**
MAXIMO (A)	-30,37	-3,1	-4,6	-18,37
MEDIA(A)	-16,27	-2,9	-0,4	-5,4
MEDIANA(A)	**-15,83**	**-7,7**	**0,43**	**4,27**

Tabla 13. Porcentajes de mejora/empeoramiento con respecto al mínimo.

Como se puede observar, la única opción que mejora y muy poco es en las palabras *socks* (0,4%) y *boots* (4,3%) cuando se usa la mediana en ambos casos, sin embargo esta mejora queda anulada con el empeoramiento cometido en las palabras *jeans* (-15,83) y *house* (-7,7). En los demás casos, los resultados son mayores que los obtenidos con el mínimo y en ocasiones la diferencia llega a ser muy significativa como en el caso del máximo con *jeans* (-30,4%). Por eso a partir de aquí se procederá a utilizar el mínimo en el análisis de los resultados.

- **Comparación de resultados entre base de datos de un locutor y base de datos de varios locutores:**

A continuación se compara la fiabilidad del reconocimiento con un locutor interno femenino (femenino 1), cuando este pronuncia las 4 palabras anteriores contra 2 bases de datos, una grabada por él mismo y otra con 4 locutores entre los que también se encuentra él. Los 4 locutores son dos femeninos y dos masculinos.

En la tabla 14 se muestran los resultados medios obtenidos cuando el locutor interno femenino, pronuncia cada palabra un número de 10 intentos usando los dos tipos de bases de datos. Al final de la tabla, se ha calculado el porcentaje de mejora o empeoramiento para cada palabra al usar la base de datos A y B. Este porcentaje está calculado tomando como referencia la base de datos que tiene 4 palabras grabadas (A).

BASE DATOS	**LOCUTOR**	**JEANS**	**HOUSE**	**SOCKS**	**BOOTS**
Locut. interno,4 (A)	INT:FEM	0,737	0,762	0,88	0,874
4locutores,16(B)	INT:FEM	0,799	0,867	0,855	0,814
Mejora(+)/Empeora	100*(A-B)	-6,2	-10,5	**2,5**	**6**

Tabla 14. Comparativa de bases de datos de 4 y 16 palabras con locutor interno femenino.

A la vista de los resultados al usar la base de datos creada por un solo locutor (A), se obtiene una leve mejora en la fiabilidad en cuanto a *socks* (2,5%) y *boots* (6%). Sin embargo la lógica dice que debería producirse un empeoramiento en todos los casos, ya que en la base de 4 locutores (B) hay grabaciones que no han sido realizadas por el locutor interno femenino y por tanto habría un mayor error en el reconocimiento.

Existen 2 razones por las cuales se produce una mejora en la fiabilidad del reconocimiento en los casos de *socks* y *boots*:

- Que el numerador en el cálculo de la relación usando la base de datos A sea mayor que el obtenido de la base de datos B.

  En las tablas 15a y 16a se muestran las puntuaciones y relación asociadas al quinto intento que realizó el locutor femenino1 con la base de datos A, pronunciando *socks*. Sin embargo las tablas 15b y 16b muestran los datos asociados al sexto intento de dicho locutor contra la base de datos B, pronunciando *boots*.

  Por ejemplo, como podemos observar en la tabla 15a, el numerador de la relación es 436,92 (primera fila y tercera columna) mientras que en la tabla 16a, el numerador de la relación a calcular será 371,66 (primera fila y tercera columna). El numerador de la base de datos A es mayor que el de la base de datos B. Lo mismo sucede en las tablas 15b y 16b, sin embargo en este caso la palabra a analizar fue *boots*.

BASE DE DATOS DE 4 PALABRAS (A)										
SE PRONUNCIA SOCKS						SE PRONUNCIA BOOTS				
JEANS	HOUSE	SOCKS	BOOTS	RELACION		JEANS	HOUSE	SOCKS	BOOTS	RELACION
458,96	679,75	**436,92**	593,08	0,95197839		370,34	825,1	567,9	**309,84**	0,8366366
…	…	…	…	…		…	…	…	…	…
				0,8797874						**0,8741319**
		(a)					(b)			

Tabla 15.Puntuaciones de *socks* y *boots* usando base de datos de 4 palabras (A).

BASE DE DATOS DE 16 PALABRAS (B)										
SE PRONUNCIA SOCKS						SE PRONUNCIA BOOTS				
JEANS	HOUSE	SOCKS	BOOTS	RELACION		JEANS	HOUSE	SOCKS	BOOTS	RELACION
458,96	679,75	**371,66**	420,23	0,88442044		370,34	825,1	339,8	**257,14**	0,75673926
…	…	…	…	…		…	…	…	…	…
				0,85452913						**0,8144766**
		(a)					(b)			

Tabla 16. Puntuaciones de *socks* y *boots* usando base de datos de 16 palabras (B).

- Que el denominador para calcular las relaciones usando la base de datos A sea menor que el denominador usado en la base de datos B.

Por ejemplo en la tabla 17a y 17 b mostramos un caso específico obtenido al pronunciar *socks* por el locutor femenino1. El denominador usado en la base de datos A es 390,06 (primera fila y

primera columna de tabla 17a) mientras que el de la base de datos B es 420,23 (primera fila y cuarta columna de tabla 17b), siendo por tanto el de A menor.

BASE DE DATOS DE 4 PALABRAS (A)					BASE DE DATOS DE 16 PALABRAS (b)				
SE PRONUNCIA SOCKS					SE PRONUNCIA SOCKS				
JEANS	HOUSE	SOCKS	BOOTS	RELACION	JEANS	HOUSE	SOCKS	BOOTS	RELACION
390,06	679,75	371,66	593,08	0,9529743	458,96	679,75	371,66	**420,23**	0,88442044
...	...	...	...	...	...	...	...	...	...
				0,8797874					**0,85452913**
			(a)					(b)	

Tabla 17. Puntuaciones al pronunciar *socks* por locutor femenino1 usando base de datos A y B.

En este caso, no se puede concluir definitivamente el uso de una base de datos o de otra, por eso se llegará a una conclusión final cuando se expliquen las siguientes pruebas.

Un locutor externo femenino pronuncia 4 palabras varias veces usando la base de datos de 4 (A) y de 16 palabras (B), el resultado de calcular las relaciones correspondientes se muestra en la tabla 18.

BASE DATOS	LOCUTOR	JEANS	HOUSE	SOCKS	BOOTS
Locut. interno,4 (A)	EXT:FEM	0,945	0,754	1,065	1,184
4locutores,16 (B)	EXT:FEM	0,919	0,702	1,024	0,65
Mejora(+)/Empeora	100*(A-B)	2,6	5,2	4,1	53,4

Tabla 18. Comparativa de bases de datos de 4 y 16 palabras con locutor externo femenino.

En todas las palabras se produce una mejora en la calidad del reconocimiento. Siendo la más significativa la correspondiente a la palabra *boots*. La gran mejora que se produce en esta palabra hace que el locutor externo femenino pase de equivocarse en la primera base de datos, obteniendo una relación mayor que uno (primera fila y cuarta columna: 1,184) a no equivocarse en la segunda teniendo una relación mucho menor que uno (segunda fila y cuarta columna: 0,65).

También es necesario destacar que en las pruebas individuales de la palabra *socks* el reconocedor se equivoca un número elevado de veces dando lugar a una relación superior a 1, concretamente se equivoca 6 veces usando la base de datos B y todas las veces usando la base de datos A. Esta relación no baja de 1 en ninguno de los dos tipos de bases de datos.

En esta prueba la conclusión que se obtiene es que el reconocedor comete menos errores cuando funciona con locutores externos femeninos, si en la base de datos de grabaciones hay diversidad en cuanto a locutores se refiere.

Seguidamente en la tabla 19 se mostrarán los resultados extraídos de las pruebas realizadas por un locutor externo masculino, que igual que los anteriores pronunció al micrófono 10 veces cada una

de las palabras propuestas. Para el cálculo de mejora/empeoramiento usamos como referencia la base de datos de 4 palabras:

BASE DATOS	LOCUTOR	JEANS	HOUSE	SOCKS	BOOTS
Locut.interno,4 (A)	EXT:MASC	0,788	0,934	0,986	1,236
4locutores,16 (B)	EXT:MASC	0,972	0,871	0,904	0,821
Mejora(+)/Empeora	100*(A-B)	-18,4	6,3	8,2	41,5

Tabla 19. Comparativa de bases de datos de 4 y 16 palabras con locutor externo masculino.

En 3 de las 4 palabras se produce una mejora en el reconocimiento y concretamente en el caso de la palabra *boots* esa mejora es muy significativa, se pasa de cometer un total de 5 errores a cometer sólo 2 dentro de los 10 intentos de reconocimiento de la palabra *boots*.

Sin embargo en la palabra *jeans* el reconocimiento empeora un 18,4% que puede ser debido a dos motivos:

- Las puntuaciones obtenidas por el reconocedor cuando se usa la base de datos de 4 palabras proporcionaban un mayor margen de error entre el acierto y el fallo del reconocimiento, dando lugar a una relación más pequeña comparando con la base de datos de 16. Se puede observar este hecho en la tabla 20.

  Por ejemplo si nos fijamos en la primera fila y primera columna de la base de datos de 4 palabras, el valor 399,96 sería el numerador de la relación, mientras que el denominador sería 571,96 que está en la primera fila y tercera columna. Sin embargo el valor de puntuación del denominador en la base de datos de 16 sería 402,24, en la primera fila y cuarta columna. Al disminuir el denominador se acerca más al valor de puntuación del numerador y la relación se hace más grande usando la base de datos de 16.

- Al existir un mayor número de palabras en la segunda base de datos, las similitudes entre las mismas se hacen un poco mayores y por tanto las puntuaciones obtenidas en los intentos de reconocimiento serán más pequeñas. Esto influye en el resultado de la relación, ya que el valor del denominador será más pequeño y dará lugar a una relación más grande. En la tabla 20 se muestra un ejemplo de lo explicado.

  Por ejemplo si comparamos las columnas de *boots* se ve que las puntuaciones son menores al usar la base de datos de 16 palabras. Lo mismo sucede con las columnas de *jeans*, *house* y *socks*.

SE PRONUNCIA JEANS, BBDD 4 PALABRAS						SE PRONUNCIA JEANS, BBDD 16 PALABRAS				
JEANS	HOUSE	SOCKS	BOOTS	RELACION		JEANS	HOUSE	SOCKS	BOOTS	RELACION
399,96	948,780	**571,96**	641,172	0,699213		**399,969**	853,205	420,623	**402,2457**	0,994340
602,217	1145,68	717,142	856,035	0,8397088		568,8398	970,4976	592,715	592,3582	0,96029z
522,467	971,247	577,705	798,981	0,9043829		522,4671	786,2007	546,981	562,5724	0,955182
375,585	968,917	562,301	619,789	0,6679429		375,5852	839,0212	441,970	403,0143	0,931940
384,59	932,607	585,489	611,839	0,6568762		384,594	856,2944	405,554	369,1013	1,041974
479,799	1037,33	636,351	733,711	0,7539851		479,7996	915,2063	464,865	478,7516	1,032124
411,970	950,112	598,190	621,241	0,6886945		411,9706	860,1831	408,169	402,2987	1,024041
657,443	1136,81	781,858	843,889	0,8408732		574,9398	961,2776	608,141	583,4337	0,985441
654,448	1204,99	730,20	895,605	0,8962516		612,5413	977,4492	640,656	637,4722	0,960891
749,982	1299,2	802,81	995,478	0,9341888		597,3054	999,5391	708,211	727,1219	0,843399
				0,788158						0,972963

Tabla 20. Cálculo de relaciones con BBDD de 4 y de 16 palabras al pronunciar jeans.

A pesar de este empeoramiento en *jeans*, el reconocedor no llega a equivocarse ya que la relación no llega a ser superior a 1.

Como conclusión de estas pruebas, se determina que la base de datos de varios locutores es más óptima cuando las personas que pronuncian las palabras no son las que previamente las han grabado. Además no se produce un empeoramiento significativo en el reconocimiento para las personas que han grabado las palabras de la base de datos, con respecto a una supuesta base de datos de referencia en la que sólo existiese la persona que hace las pruebas (tabla 14).

6.3. Grabación de 4 palabras con diferentes intensidades y posiciones usando 4 locutores

Se pretende determinar si es mejor grabar una base de datos con varios locutores y palabras duplicadas diferenciándolas por su posición dentro del tiempo de captura y por su nivel de intensidad o simplemente es válido usar la base de datos con palabras pronunciadas por varios locutores y sin duplicidades.

En las siguientes tablas de resultados, el primer tipo de base de datos es el que no tiene duplicidades (A) y el segundo el que si las posee (B). Por ejemplo, en la base de datos B se tendría grabada la palabra *jeans* un total de 16 veces: 4 tipos de *jeans* para cada uno de los 4 locutores distintos. La forma de grabación de esos 4 tipos sería la siguiente:

- Intensidad de pronunciación normal con palabra situada al principio del tiempo de captura establecido (un segundo).
- Intensidad de pronunciación normal con palabra colocada al final del tiempo de captura.
- Intensidad de pronunciación alta (sin saturar) con palabra situada al principio del segundo.
- Intensidad de pronunciación alta (sin saturar) con palabra colocada al final del segundo.

La base de datos A contiene 16 palabras grabadas y es la misma que en pruebas anteriores y la base de datos B posee 64 palabras, 16 palabras por 4 locutores. Estas 16 palabras corresponden a 4 *jeans*, 4 *house*, *4 boots* y 4 *socks*.

Es necesario comentar que como en casos anteriores el ruido ambiente grabado es una palabra adicional y las condiciones de grabación son las mismas que se han utilizado inicialmente, tiempo de captura 1 segundo y frecuencia 32768 muestras/s. Además de que los locutores que grabaron las palabras también fueron los mismos.

Los valores numéricos de las relaciones que se muestran en las tablas siguientes para el caso de base de datos A son los mismos que los usados en la prueba anterior. Los resultados de la base de datos B se han obtenido de igual forma que en otras ocasiones, cada locutor pronuncia 10 veces cada una de las palabras grabadas: *jeans*, *house*, *socks* y *boots*.Como dato relevante se ha incluido una nueva columna que indica el número de errores de reconocimiento totales que se han producido durante el proceso de evaluación, esto quiere decir por ejemplo que si la palabra reconocida debía ser *house* el reconocedor ha elegido como opción válida *jeans*.

Los resultados numéricos cuando el locutor interno femenino (femenino 1) pronuncia las palabras son las que se muestran en la tabla 21:

BASE DATOS	LOCUTOR	JEANS	HOUSE	SOCKS	BOOTS	Nº ERRORES
4locutores,16 (A)	INT:FEM	0,799	0,867	0,855	0,814	0
4locutores,64 (B)	INT:FEM	0,806	0,85	0,902	0,879	2
Mejora(+)/Empeora	100*(A-B)	-0,7	1,7	-4,7	-6,5	

Tabla 21. Comparativa de bases de datos de 16 y 64 palabras con locutor interno femenino.

El cálculo de la mejora o el empeoramiento se produce tomando como referencia la base de datos de 16 palabras (A).

En líneas generales baja el rendimiento del reconocimiento, produciendo una única mejora y muy poco significativa en la palabra *house*.En este caso el reconocedor se equivoca en más ocasiones cuando se usa una base de datos mayor. Seguidamente se muestran los resultados obtenidos de fiabilidad de reconocimiento asociados al locutor externo femenino (tabla 22):

BASE DATOS	LOCUTOR	JEANS	HOUSE	SOCKS	BOOTS	Nº ERRORES
4locutores,16 (A)	EXT:FEM	0,919	0,702	1,024	0,65	7
4locutores,64 (B)	EXT:FEM	0,702	0,867	0,763	0,859	2
Mejora(+)/Empeora	100*(A-B)	21,7	-16,5	26,1	-20,9	

Tabla 22. Comparativa de bases de datos de 16 y 64 palabras con locutor externo femenino.

En este caso se observa como usando el segundo tipo de base de datos disminuye significativamente el número de errores de reconocimiento. Además, se produce una gran mejora en el caso específico de las palabras *jeans* y *socks*. Especialmente en la palabra *socks*, pasando la relación de ser en término medio mayor que 1 a ser inferior.

Cabe destacar que empeora el reconocimiento en las otras dos palabras, *house* y *boots* un 16,5% y 20,9% respectivamente.

Finalmente se explican los resultados extraídos de las pruebas realizadas por el locutor externo masculino (tabla 23):

BASE DATOS	LOCUTOR	JEANS	HOUSE	SOCKS	BOOTS	Nº ERRORES
4locutores,16 (A)	EXT:MASC	0,972	0,871	0,904	0,821	6
4locutores,64 (B)	EXT:MASC	0,797	0,88	0,88	0,976	5
Mejora(+)/Empeora	100*(A-B)	17,5	-0,9	2,4	-15,5	

Tabla 23. Comparativa de bases de datos de 16 y 64 palabras con locutor externo masculino.

En esta ocasión no disminuyen en gran medida los errores de reconocimiento, se comete solamente un fallo menos usando la base de datos de 64 palabras.

Además, la mejora de las palabras *jeans* y *socks* se ve compensada con el empeoramiento de *house* y *boots.* Por tanto en esta ocasión no hay una gran diferencia entre usar la base de datos del primer tipo y la del segundo.

En vista de los resultados de los tres tipos de locutores, se obtiene una mejora poco significativa en el reconocimiento en comparación con la cantidad de palabras necesarias en la nueva base de datos, dado que se tendría que cuadriplicar el número de palabras con respecto al caso anterior. Esto conllevaría además un coste temporal en el resultado del reconocimiento. Por tanto se decide usar la base de datos de varios locutores y sin duplicidades, es decir la base de datos de 16 palabras, para la integración del reconocimiento de voz en nuestro videojuego.

6.4. Captura del ruido de ambiente

Se decidió capturar las puntuaciones que el asset de reconocimiento calculaba para el ruido, es decir, cuando en el micrófono no se pronunciaba nada.

El objetivo de esta prueba era determinar un umbral de ruido para mejorar el reconocimiento.

Se capturó el ruido un total de 10 veces. Y el resultado fue el que se presenta en la tabla 24:

Orden pruebas	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Puntuación *noise*	11,83	13,82	11,7	111,98	10,7	10,7	10,2	11,7	11,68	10,42

Tabla 24. Puntuación obtenida del ruido.

Como podemos observar el umbral de ruido está en unos márgenes muy pequeños salvo en una ocasión que es un valor muy elevado, 111,98, y es debido a una mala captura del reconocedor.

Finalmente gracias a la realización de estas pruebas se estableció como un umbral un valor de puntuación de 150, por debajo del cual el reconocedor actuaría como si no se hubiese pronunciado ninguna palabra. En caso contrario el reconocedor determinaría qué palabra ha sido la pronunciada por el usuario, sin tener en cuenta la puntuación de ruido

7. MANUAL DE USUARIO

7.1. Entorno de sistema requerido

Los requisitos que son necesarios para ejecutar el videojuego son:

- Sistema operativos: Windows XP o posterior; Mac OS X 10.6 o posterior.
- Tarjeta gráfica: Cualquier tarjeta de gráficos 3D disponible en el mercado.
- Tarjeta de Sonido: Cualquiera de las disponibles en el mercado para su instalación en PCs.
- Micrófono: Este dispositivo debe ir conectado a la entrada estéreo o de micrófono de la tarjeta de sonido (2,5 mm de diámetro).
- Teclado estándar.
- Ratón estándar.
- Monitor que admita resoluciones de 640x480 o superiores.

7.2 Configuración del micrófono

Como un primer paso debemos asegurarnos que nuestro ordenador lleva un micrófono incorporado o que tenemos conectado un micrófono multimedia al mismo. Una vez realizado este paso debemos configurar el micrófono a utilizar y comprobar que las características del mismo son las idóneas para una buena captura de audio. Hay que tener en cuenta las siguientes consideraciones:

- El volumen de grabación: que debe tener un valor superior a 60 e inferior a 90.

- Amplificación del micrófono: conviene incluir una ganancia al micrófono de 10 dB.
- Usar el micrófono en un ambiente lo menos ruidoso posible para ayudar a la mejora de la fiabilidad de reconocimiento.

7.3. Inicio del videojuego

Los archivos necesarios para la ejecución del videojuego se encuentran en una carpeta con los recursos necesarios junto a una archivo *.exe*.

Una vez que hemos ejecutado el archivo *.exe* nos aparece una ventana previa como la mostrada en la figura 9. En ella aparece una serie de opciones relativas a los gráficos (pestaña *Graphics*) que se quieren asumir a la hora de ejecutar el videojuego, referentes a la resolución de la pantalla y a la calidad deseada. También se presenta la opción de elegir si el videojuego será presentado en forma de ventana o no, selección denominada *Windowed*. En caso de que no se seleccione esta opción, para salir de la pantalla del videojuego será necesario presionar la tecla *Windows* del teclado.

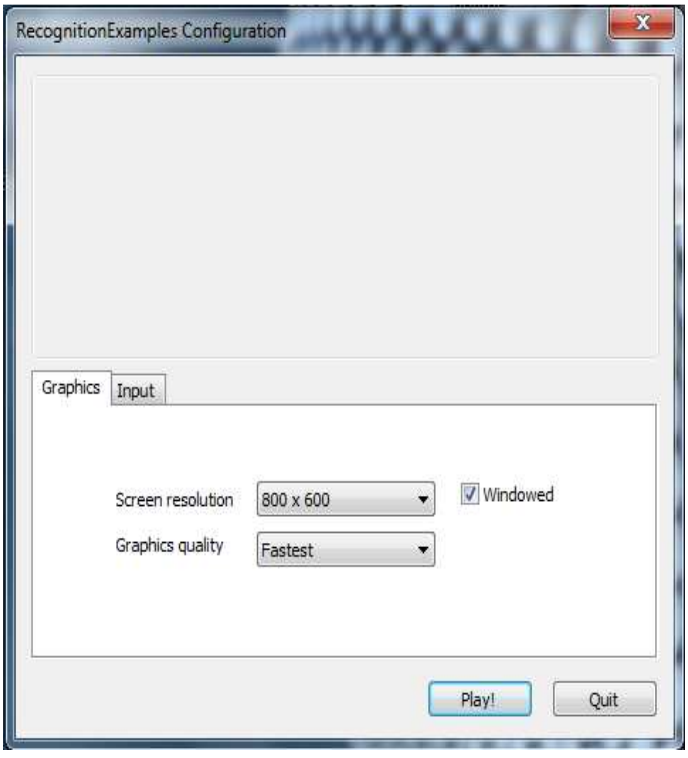

Figura 9. Ventana de configuración del videojuego.

Además, en la misma ventana de configuración del videojuego, concretamente pestaña *Input* , se encuentra información sobre qué teclas son las encargadas de controlar las acciones de juego. Se pueden cambiar los controles del videojuego a gusto del usuario antes de iniciar el mismo (figura 10).

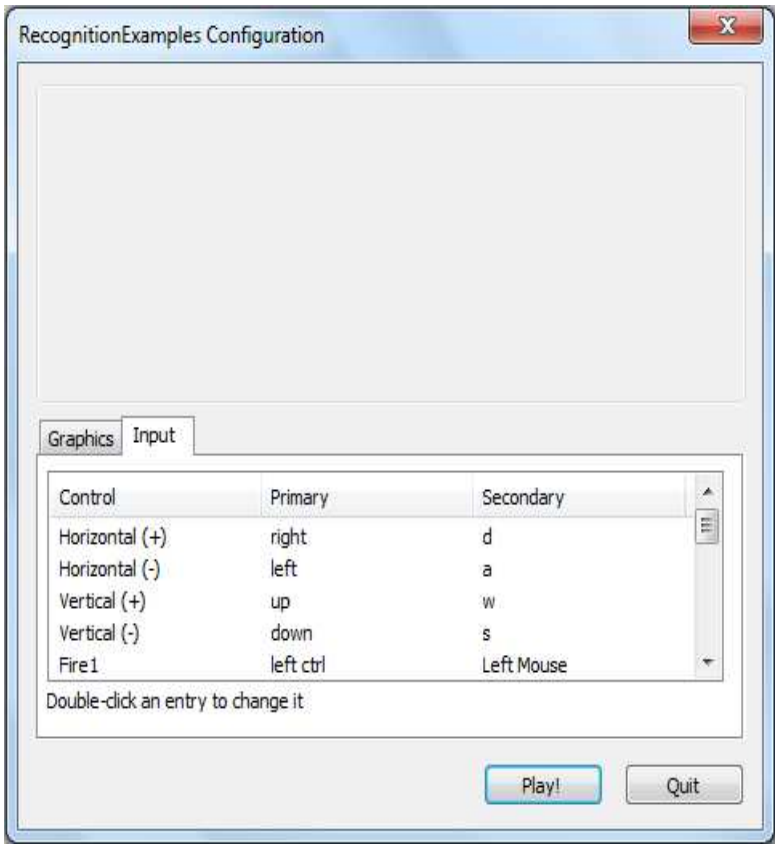

Figura 10. Controles del videojuego.

Una vez que se han establecido las características de gráficos que tendrá el videojuego y los controles, presionamos el botón *play!* y nos aparecerá la primera escena del videojuego que es la que representa la figura 11.

En esta escena se presentan diversas opciones en forma de botones. Un total de 4 botones cuyas funcionalidades se definen a continuación:

- Interrogación: para mostrar los controles del juego.
- Botón de salir: para cerrar la aplicación del videojuego.
- Selección de personaje: con el que se jugará la aventura gráfica. Un botón para elegir al marciano y otro botón para el niño.

Figura 11. Primera escena del videojuego.

Si pulsamos en la *interrogación* se nos mostrará la pantalla de ayuda del videojuego con información sobre los controles que se pueden usar durante la experiencia en el videojuego. Este panel de ayuda es mostrado en la figura 12.

Figura 12. Escena de ayuda.

Para empezar a jugar es necesaria la selección de un avatar, una vez se pulse el botón en el que aparece un *marciano* o un *niño* emergerá la siguiente escena del juego. Si se elige como personaje principal al marciano se mostrará la pantalla del juego representada por la figura 13. Para mover al personaje hacia la derecha y la izquierda se usarán las flechas del teclado *right* y *left*, además de la barra espaciadora para efectuar saltos.

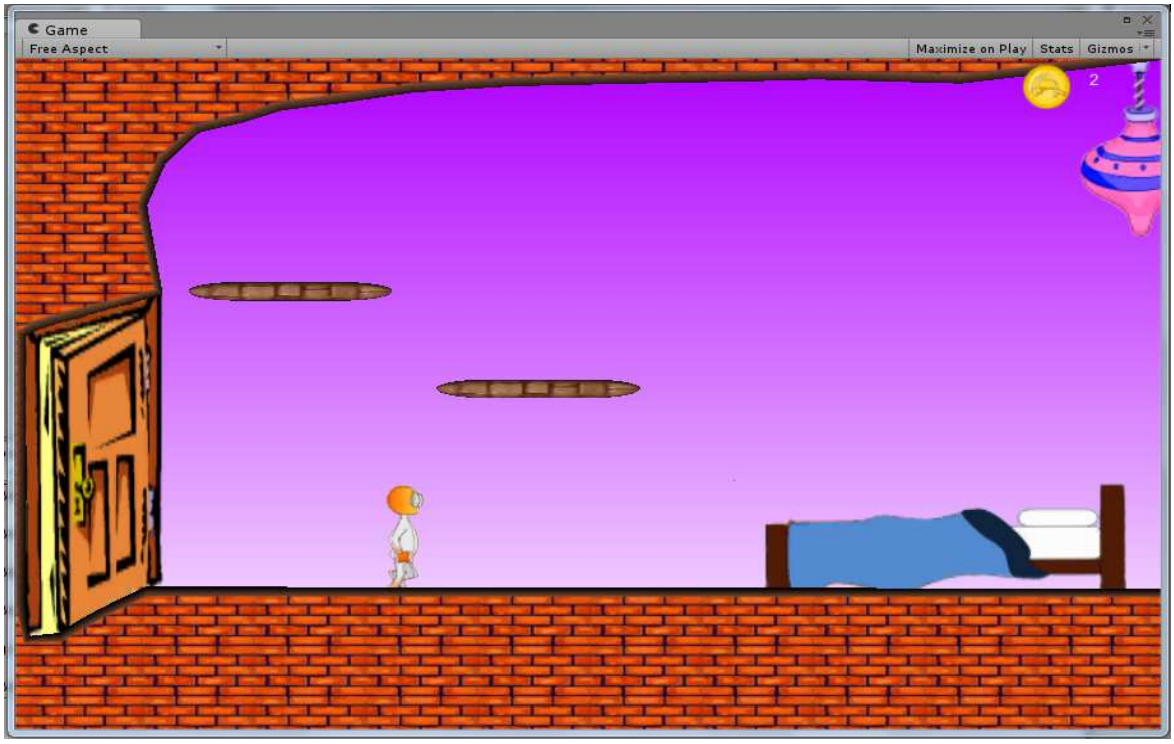

Figura 13. Escena 2 con avatar de marciano.

A continuación hay que buscar la forma con la cual pasar al escenario siguiente, y eso se consigue tocando al monstruo que aparecerá en algún lugar de la pantalla. En la figura 14 se observa el monstruo que permite cargar la tercera y última escena del videojuego.

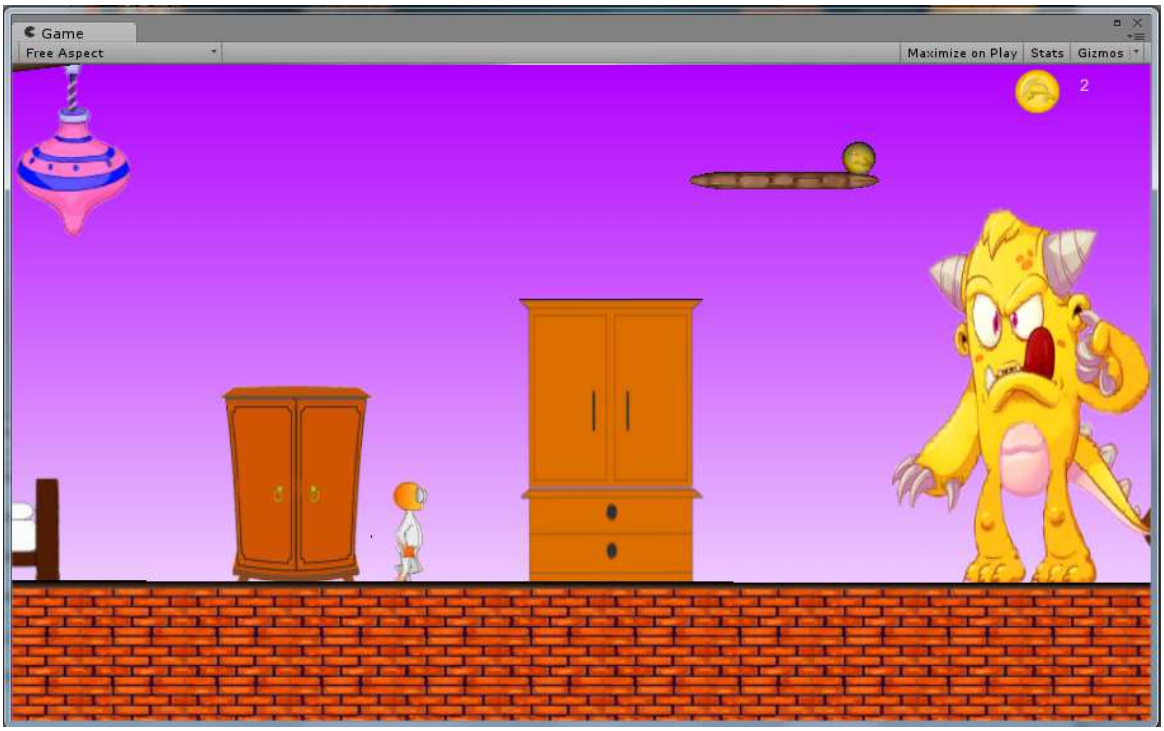

Figura 14. Continuación de la escena 2.

Con el objetivo de distinguir el resultado de haber seleccionado el otro personaje disponible en la pantalla inicial (el niño), se procede a mostrar el efecto en la figura 15.

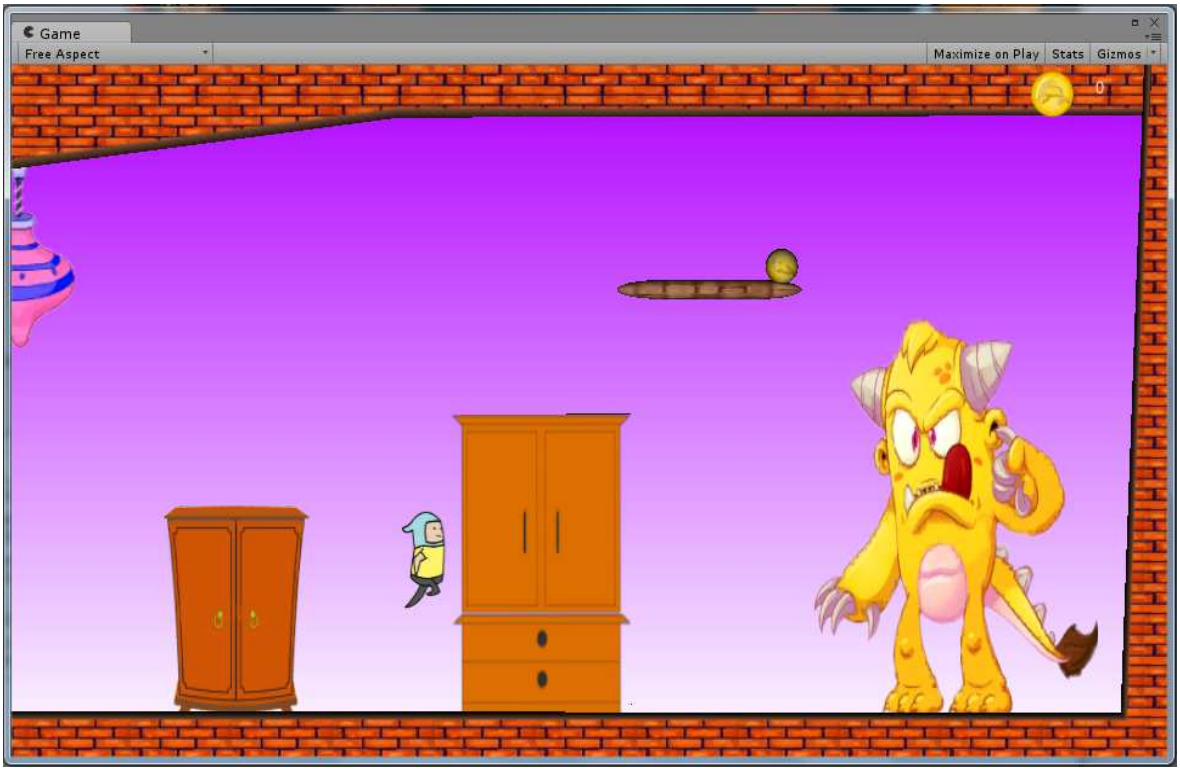

Figura 15. Escena 2 con cambio de personaje.

Una vez se ha tocado el monstruo en el escenario 2 del videojuego, se carga la siguiente pantalla, representada por la figura 16.

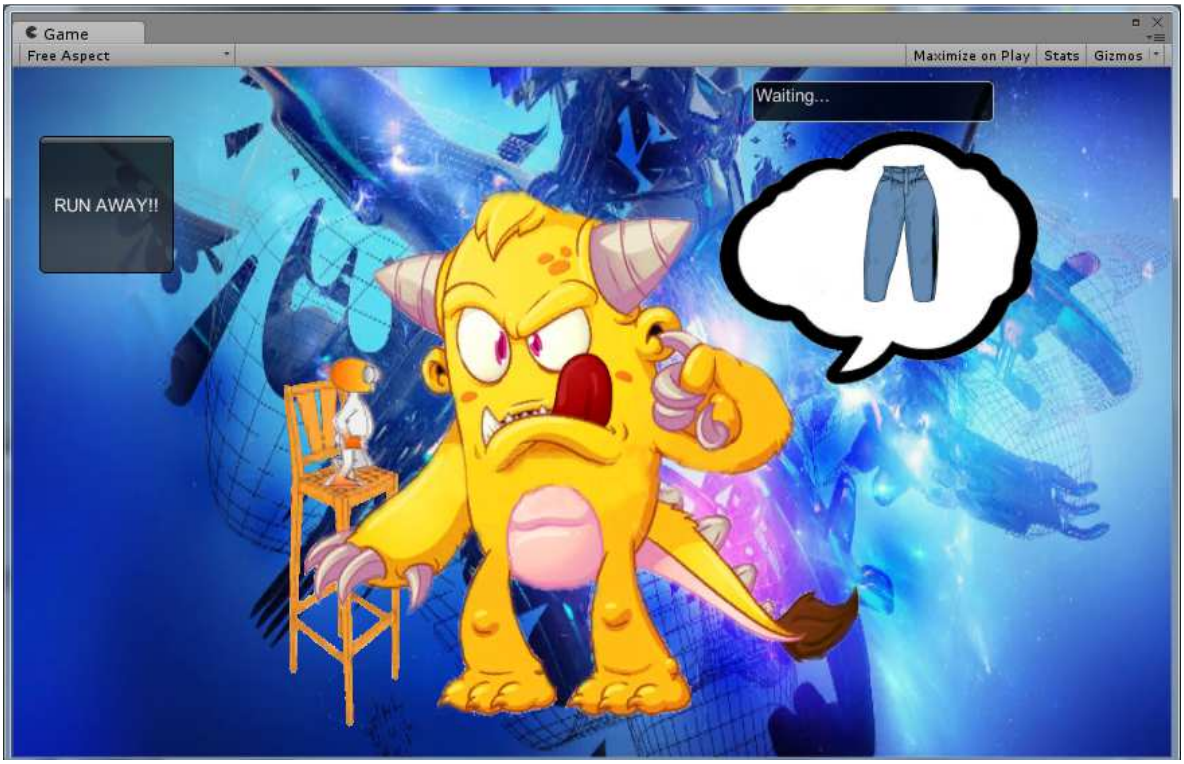

Figura 16. Escena 3 de reconocimiento de voz.

En la última pantalla del videojuego, el usuario debe pronunciar al micrófono el nombre del objeto que aparezca representado por el pensamiento del monstruo, y nada más terminar de pronunciar la palabra debe presionar el botón que aparece en la esquina superior izquierda con el nombre de *RUN AWAY!*. Una vez realizado este paso, precederá una animación que permitirá al usuario conocer el resultado de reconocimiento de voz de la palabra pronunciada anteriormente además de volver a la escena anterior. Si el usuario ha pronunciado bien el nombre del objeto visible en pantalla se incrementará la puntuación de su marcador en un punto y saldrán monedas en la pantalla anterior que podrá recoger, si por el contrario se ha pronunciado mal disminuirá su puntuación también en un punto. El objeto mostrado en este escenario será el mismo durante 3 ocasiones, si en las 3 ocasiones el usuario pronuncia el nombre mal, se cambiará la opción a mostrar.

8. CONCLUSIONES Y LÍNEAS FUTURAS

8.1. Conclusiones

La enseñanza virtual está evolucionando a gran velocidad debido al incremento de uso de ordenadores, teléfonos móviles y cualquier otro dispositivo similar a estos. Hay que estar en la ola de la evolución de este tipo de técnicas de enseñanza, para mantener una educación renovada y actual, a la vez que atractiva y entretenida para los usuarios. Las ventajas de esta forma de educación son abundantes y variadas: permiten almacenar el progreso del usuario a lo largo de su proceso de aprendizaje, proporcionan una comunicación constante entre usuario e instructor y aumentan el nivel de implicación del usuario con respecto a otras alternativas de aprendizaje, ya que es un tipo de enseñanza atractiva y motivadora.

La enseñanza virtual se sustenta en diferentes plataformas online, aplicaciones u otros recursos software. Este proyecto se situó en el campo de la plataforma del videojuego para mejorar el aprendizaje de los usuarios en un cierto conocimiento, la lengua inglesa. El objetivo principal fue desarrollar un videojuego para mejorar el conocimiento de esta lengua para usuarios no nativos en la misma. Concretamente el proyecto se focalizó en mejorar la expresión oral en la lengua inglesa, integrando un reconocimiento de voz. Una vez se tuviera terminado este videojuego, se integraría en un proyecto más grande, un videojuego que ayudaría a aprender y a conocer todas las destrezas de la lengua inglesa, creando un método de enseñanza completo.

Por otro lado, se pensó que el nivel más adecuado de los ejercicios de expresión oral sería un nivel destinado a usuarios que se encontrasen en el 2º ciclo de educación primaria (6º primaria), porque están en una edad a caballo entre la adolescencia y la niñez. Siguen encontrando interesante el uso de videojuegos, aunque es necesario un juego más serio para motivarles y animarles a aprender. Los ejercicios del videojuego actuarían como un complemento a la enseñanza de la lengua inglesa proporcionada en las aulas. Sería utilizado tanto por profesores como por los alumnos y mediante su uso se detectarían posibles carencias que podrían tener los usuarios.

Para cumplir y resolver los objetivos que se habían propuesto se tomaron un conjunto de decisiones. La primera decisión tomada estuvo en relación con la forma de desarrollo del

videojuego, se realizó la elección del entorno de programación más óptimo, que como se explicó anteriormente fue Unity®. Una vez se tomó esta decisión, se procedió a seleccionar la librería de reconocimiento de voz que mejor se ajustaba a las necesidades de los ejercicios del videojuego, sin embargo después de haber elegido una resultó que no podía integrarse en el entorno de programación escogido por incompatibilidades del software y se tuvo que cambiar la elección. Finalmente se usó como librería de reconocimiento la llamada *WordDetection*, adquirida en la tienda online de Unity®.

Una vez que se tenían las alternativas seleccionadas se procedió a diseñar y programar el videojuego, integrando correctamente la función de reconocimiento vocal. Como comprobación de la fiabilidad de esta funcionalidad, se realizaron diversas pruebas con distintos objetivos.

Las primeras pruebas realizadas fueron destinadas a seleccionar las características de grabación de las palabras de la base de datos del reconocedor, se concluyó que el tiempo de grabación más óptimo era 1 segundo y la frecuencia a utilizar 32768 muestras/segundo.

A continuación se hicieron pruebas para determinar si era conveniente tener grabaciones repetidas de las palabras de la base de datos por diferentes locutores para mejorar la calidad de reconocimiento de voz. Los resultados de esta hipótesis de trabajo fueron que mejoraba la fiabilidad del reconocedor de voz en un porcentaje considerable. Concretamente los locutores que grabarían las palabras serían dos femeninos y dos masculinos.

La última prueba a mencionar fue la que incluía repeticiones extra de palabras en la base de datos con diferentes intensidades y desplazamientos dentro del tiempo de grabación. En esta prueba se cuadriplicó el tamaño de la base de datos, incrementándose el tiempo de carga del videojuego por tener que recuperar más perfiles de palabras del registro del sistema. El objetivo de la prueba era determinar si mejoraba aun más el reconocimiento. El resultado que se obtuvo fue que la poca mejora obtenida en la fiabilidad no compensaba el hecho de tener que cuadriplicar la base de datos de grabaciones.

La validez del sistema de reconocimiento de voz es aceptable, sin embargo conviene mejorar aún más la fiabilidad del mismo mediante la mejora en las técnicas de comparación de palabras definidas en la librería de reconocimiento elegida, *WordDetection*. El cálculo de los espectros de las palabras es muy deficiente, no llegando a normalizar las palabras en cuanto a energía se refiere para poder realizar una comparación eficiente. Sin realizar los cambios necesarios en las librerías, se podría engañar al reconocedor de voz emitiendo sonidos sonoros parecidos a la pronunciación de una palabra y el reconocedor daría como válida una pronunciación que no se ajusta a la realidad.

8.2. Líneas futuras

En base a los problemas detectados se proponen diferentes alternativas para solucionarlos.

1. Integrar el reconocimiento en el juego completo.

2. Grabación de las palabras con herramientas adecuadas y profesionales: actualmente se han usado herramientas convencionales de grabación que no ofrecen la suficiente robustez, en contra del ruido de ambiente.

3. Probar más librerías de reconocimiento, incluso no gratuitas.

4. Se pretende en un futuro realizar una librería de reconocimiento de voz propio que incluya las siguientes mejoras:

   Energía: métodos para controlar la energía de las palabras pronunciadas al micrófono para que la variación de intensidad con la que se pronuncian no afecte al resultado dado por el reconocedor.

   Modelos ocultos de Markov: emplear métodos que implementen las características de los modelos ocultos de Markov para el reconocimiento de voz [57] [58].

5. Evaluar el aprendizaje real de los alumnos que usarían el videojuego según decían los autores de [5].

.

9. REFERENCIAS

Estado del arte

[1] Tecnologías de la Información y la Comunicación para la Innovación Educativa. E.R. Velasco Sánchez. 1ª edición. Editorial: Díaz de Santos, México 2012. ISBN: 978-84-9969-609-6

[2] Reviewing the need for gaming in education to accommodate the Net Generation. G. Bekebrede, H.J.G. Warmelink, I.S. Mayer. Computers & Education vol. 57 pp. 1521-1529. 2011.

[3] How can exploratory learning with games and simulations within the curriculum be most effectively evaluated? S. de Freitas, M. Oliver. Computers & Education vol. 46 pp. 249-264. 2006.

[4] Digital game-based learning: towards an experiential gaming model. K. Kiili. The Internet and Higher Education vol. 8 pp. 13-24. 2005.

[5] Serious games as new educational tools: how effective are they? A meta-analysis of recent studies. C. Girard, J. Ecalle, A. Magnant. Journal of Computer Assisted Learning vol. 29 pp. 207-219. 2013.

[6] Profiling the Educational Value of Computer Games. A. Frazer, A. Recio-Saucedo, L. Gilbert, G. Wills. Interaction Design and Architecture(s) Journal vol. 19 pp. 9-27. 2013.

[7] Psychological Perspectives on Motivation through Gamification. M. Sailer, J. Hense, H. Mandl and M. Klevers. Interaction Design and Architecture(s) Journal vol. 19 pp. 28-37. 2013.

[8] The effects of video game playing on attention, memory, and executive control. W.R. Boot, A.F. Kramer, D.J. Simons, M. Fabiani, G. Gratton. Acta Psychologica vol. 129 pp. 387-398. 2008.

[9] Education by plays and games. G. Ellsworth Johnson. The Athenaeum Press (Gin&Company). Boston (USA). 1907.

[10] Serious Games: Games That Educate, Train, and Inform. 1º edición. Michael D., Chen S. Editorial: Course Technology PTR. 2006. ISBN-10: 1-59200-622-1 ISBN-13: 978-1-59200-622-9.

[11] Constructivism: A theory of knowledge. Bodner, G.M. Journal of Chemical Education 63(10), 873-878. 1986.

[12] Origins of a behaviourist. Skinner, B.F. Psychology Today, 22–33. 1983.

[13] The role of tutoring in problem solving. Wood, D., Bruner, J.S., Ross, G. Journal of Psychology and Psychiatry, 17. 1976.

[14] The Same, But Different: The Educational Affordances of Different Gaming Genres. Frazer, A., Argles, D. and Wills, G. En: ICALT 2008: The 8th IEEE International Conference on Advanced Learning, 1 to 5 July, Spain. 2008.

[15] Investigating the impact of video games on high school students' engagement and learning about genetics. L.A. Annetta, J. Minogue, S.Y. Holmes, M.T. Cheng. Computers & Education vol. 53 pp. 74-85. 2009.

[16] Teachers' Perceptions of the Use of Computer Assited Language Learning to Develop Chilcren's Reading Skills in English as a Second Language in the United Arab Emirates. Hamed Mubarak Al-Awidi, Sadiq Abdulwahed Ismail. Spriger Science Business Media.vol.42 pp. 29-37. 2012.

[17] "International survey of the experience and perceptions of teachers" in Serious games in education-A global perspective. S. Egenfeldt-Nielsen, B.H. Sorensen, B.Meyer. Aarhus University Press. Aarhus (Dinamarca). 2011.

[18] REAP.PT Serious Games for learning Portuguese. A. Silva, C. Marqués, J. Baptista, A. Ferreira Jr., N. Mamede. Lecture Notes in Computer Science vol. 7243 pp. 248-259. 2012.

[19] Serious use of a serious game for language learning. W.L. Johnson. International Journal of Artificial Intelligence in Education vol. 20 pp. 175-195. 2010.

[20] Learning a foreign language in a mixed-reality environment. M.B. Ibáñez, C. Delgado Kloos, D. Leony, J.J. García Rueda, D. Maroto. IEEE Internet Computing vol. 15(6) pp. 44-47. 2011.

[21] Serious game motivation in an efl classroom in Chinese primary school. R. Anyaegbu, W. Ting, Y. Li. Turkish Online Journal of Educational Technology vol. 11 pp. 154-164. 2012.

[22] A serious game for second language acquisition in a virtual environment. M. Amoia, T. Bretaudiere, A. Denis, C. Gardent, L. Pérez-Beltrachini. Systemics, Cybernetics and Informatics vol. 10 pp. 24-34. 2012.

[23] Game-based language learning for pre-school children: a design perspective. B. Meyer. The Electronic Journal of e-Learning vol. 11(1) pp. 39-48. 2013.

[24] Learn English Kids. [En línea] <http://learnenglishkids.britishcouncil.org/en/> British Council. [Consulta: 31-Julio-2014].

[25] Pulitzer. [En línea] Play and learn English. <https://macmillan-pulitzer.com/?l=es.> MacMillan. [Consulta: 31-Julio-2014].

[26] Playenglish. [En línea] <http://es.playstation.com/psp/games/detail/item275177/Play English>. Sony. [Consulta: 31-Julio-2014].

[27] The NativeAccentTM pronunciation tutor: measuring success in the real world. Maxine Eskenazi, A. Kennedy, C. Ketchum, R. Olszewski, G. Pelton. Speech and Language Technology in Education (SLaTE2007) October 1-3 pp. 1-4. 2007.

[28] Virtual Dialogues with Native Speakers: The Evaluation of an Interactive Multimedia Method. William G. Harless, Marcia A. Zier, Robert C. Duncan. Calico Journal.vol. 16 pp.313-337. 1999

[29] Preliminary Tests of Language Learning in a Speech-Interactive Graphics Microworld. V. M. Holland, J. D. Kaplan, and M. A. Sabol Calico Journal vol.16 pp.339-359. 1999

[30] Gartner Symposium. Gamification 2020: What Is the Future of Gamification? . Centre Convencions Internacional Barcelona. November 5-8, 2012.

Desarrollo de videojuegos

[31] Simple DirectMedia Layer [En línea] <http://wiki.libsdl.org>. Latinga S. [Consulta 31-Julio-2014]

[32] Pygame [En línea]. <http://www.pygame.org>. Shinners P. [Consulta 31-Julio-2014]

[33] SFML, Simple and Fast Multimedia Library [En línea]. <http://www.sfml-dev.org/> Gomila L. [Consulta 31-Julio-2014]

[34] Using open source libraries in cross platform games development. Fahy, R.,Nui Galway, Krewer, L. En: Games Innovation Conference (IGIC), IEEE International. Rochester, NY: IEEE, 2012. ISBN: 978-1-4673-1359-9

[35] Allegro, a game programming library [En línea]. <http://alleg.sourceforge.net/>. Allegro developers. [Consulta: 31-Julio-2014]

[36] Información LIBGDX [En línea] <http://libgdx.badlogicgames.com/>. Mario Zechner. [Consulta: 31-Julio-2014]

[37] Marmalade [Enlínea] < http://docs.madewithmarmalade.com>. Marmalade Technologies Limited. [Consulta 31-Julio-2014]

[38] Microsoft: Next Generation of Games Starts With XNA [En línea] <https://www.microsoft.com/en-us/news/press/2004/mar04/03-24xnalaunchpr.aspx> Microfost. [Consulta: 31-Julio-2014]

[39] Web de Angengine [En línea] <http://www.andengine.org/> Nicolas Gramlich. [Consulta: 31-Julio-2014]

[40] Web Blender [En línea] <http://www.blender.org/features/>. Ton Roosendaal. [Consulta: 31-Julio-2014]

[41] Unity Spain [En línea] <http://www.unityspain.com/>. Andrés Gil, Anderson Sánchez. [Consulta 31-Julio-2014]

[42] Unity [En línea] <http://unity3d.com/>. Karsten Nielsen. [Consulta 31-Julio-2014]

[43] Using Unity 3D to facilitate mobile augmented reality game development. Sung Lae Kim, Suwon, Hae Jung Suk, Jeong Hwa Kang, Jun Mo Jung. En: Internet of Things (WF-IoT), IEEE World Forum on. Seoul: IEEE, 2014.

[44] Ogre3D [En línea] <http://www.ogre3d.org/about/features>. Assaf Raman, Jim Buck, Dave Rogers. [Consulta: 31-Julio-2014]

Reconocimiento de voz

[45] Microsoft speech SDK [En línea] <http://msdn.microsoft.com/en-us/library/hh362943.aspx>. Microsoft Corporation. [Consulta 31-Julio-2014]

[46] Dragon NaturallySpeaking SDK Client Edition [En línea] http://www.nuance.com/for-developers/dragon/client-sdk/index.htm> Nuance. [Consulta 31-Julio-2014]

[47] Nuance-speechmagic [En línea] <https://www.citrix.com/ready/en/nuance-communications/nuance-speechmagic>. Abuse management. [Consulta 31-Julio-2014]

[48] CMUsphinx [Enlínea] <http://cmusphinx.sourceforge.net/wiki/download/>. Carnegie Mellon University. [Consulta 31-Julio-2014]

[49] JuliusCode [En línea] <http://julius.sourceforge.jp/en_index.php>. Julius Development Team. [Consulta 31-Julio-2014]

[50] Recent Development of Open-Source Speech Recognition Engine Julius. Lee, Akinobu; Kawahara, Tatsuya. Proceedings: APSIPA ASC 2009: Asia-Pacific Signal and Information Processing Association. Annual Summit and Conference. Singapore, pp 131-137. 2009.

[51] Web Hidden Markov Model Toolkit. [En línea] <http://htk.eng.cam.ac.uk/>. Cambridge University Engineering Department (CUED). [Consulta 20-06-2014]

[52] Voce: Open Source Speech Interaction [En línea] <http://voce.sourceforge.net/>. Tyler Streeter. [Consulta 31-Julio-2014]

[53] SDKs, Plugins, & Tools | AT&T Developer Program [En línea] <https://developer.att.com/sdks-plugins?api=speech> . AT&T. [Consulta 31-Julio-2014]

[54] Unity Asset store. [En línea] <https://www.assetstore.unity3d.com/en/#!/content/4518>. Unity. [Consulta 31-Julio-2014]

[55] Diagramas UML online: creately <https://creately.com/app/?tempID=h165rwt81#>. Creately developers. [Consulta 31-Julio-2014]

[56] Requisitos del sistema. [En línea] <https://unity3d.com/es/unity/system-requirements>. Unity. [Consulta 31-Julio-2014]

Líneas futuras

[57] Servidores vocales interactivos: desarrollo de un servicio de páginas blancas por teléfono con reconocimiento de voz proyecto idas (Interactive telephone-based Directory Assistance Service). San-Segundo R., Colás J. , Montero J.M., Córdoba R., Ferreiros J., Macías-Guarasa J., Gallardo A., Gutiérrez J.M., Pastor J., Pardo J.M. Madrid: Universidad Politécnica Madrid.

[58] Implementación de un reconocedor de palabras aisladas dependiente del locutor. César San Martín S., Roberto Carrillo A., Revista Facultad de Ingeniería, U.T.A. (Chile), vol. 12, pp. 9-14. 2004

Documentación adicional

Microsoft C#: Curso de programación. Fco. Javier Ceballos. 2º edición. Madrid. Editorial RA-MA 2010.ISBN: 978-84-7897-986-8.

Visual Studio 2010, .NET 4.0 y ALM. Bruno Capuano. Editorial Krasis Press. ISBN: 978-84-936696-4-5.

Cómo programar en C++. Deitel. 6ºedición.Editorial Prentice Hall. ISBN: 978-97-026127-3-5.

Unity diseño y programación de videojuegos. N. Arriola Landa. Edición 2013. Editorial Fox Andina. ISBN: 978-98-718578-1-4

C# guía total del programador. N. Arriola Landa. Edición 2010. Editorial Fox Andina. ISBN: 978-98-726013-5-5

Windows speech recognition programming. J. Keith . Edición 2004. Editorial IUniverse. ISBN:0-595-30843-0